DOCUMENT RESUME

ED 403 727 EC 305 303

AUTHOR	Brown, Carrie; And Others
TITLE	The Sound-to-Speech Translations Utilizing Graphics Mediation Interface for Students with Severe Handicaps. Final Report.
INSTITUTION	Arc, Arlington, TX.
SPONS AGENCY	Department of Education, Washington, DC.
PUB DATE	[92]
CONTRACT	H180P90015
NOTE	321p.
PUB TYPE	Reports - Descriptive (141)
EDRS PRICE	MF01/PC13 Plus Postage.
DESCRIPTORS	*Augmentative and Alternative Communication; Communication Disorders; *Computer Graphics; *Computer Interfaces; Computer Oriented Programs; Input Output Devices; Mental Retardation; Neurological Impairments; Physical Disabilities; Research Projects; *Severe Disabilities; Speech Synthesizers
IDENTIFIERS	Automatic Speech Recognition; *Voice Recognition

ABSTRACT

This final report describes activities and outcomes of a research project on a sound-to-speech translation system utilizing a graphic mediation interface for students with severe disabilities. The STS/Graphics system is a voice recognition, computer-based system designed to allow individuals with mental retardation and/or severe physical disabilities (such as quadriplegia, spinal cord injury, cerebral palsy, and assorted neurological disorders) to communicate with others and control their environment. Operation of the STS/Graphics system is through vocalization, switch closure, or keyboard input for activation of electrical appliances and/or digitized speech output. Choices available to the user are displayed on a monitor as customized photographic-quality symbols representing familiar items/persons in the user's environment. The project resulted in four major outcomes: (1) development and refinement of the STS/Graphics system; (2) application, evaluation, field testing, and research results of the STS/Graphics system with students having mental retardation; (3) development of training strategies and a training manual on the use of the STS/Graphics system; and (4) documentation of the STS/Graphics system research with two videotapes. Individual sections of the report present the project's objectives, methodology, and a description of hardware and software used. Also attached are the training manual, sample evaluation forms, and the following papers or articles by Carrie Brown and others: "Research Focusing on Freedom of Choice, Communication, and Independence Using Eyegaze and Speech Recognition Assistive Technology"; "The Sound-to-Speech Translation Utilizing Graphic Symbols"; "Speech Recognition and Graphics Research for Persons Having Mental Retardation"; and "The Sound-to-speech Translation System Using Photographic-Quality Graphic Symbols." (Individual papers contain references.) (Author/DB)

The Sound-to-Speech Translations Utilizing Graphics Mediation Interface for Students with Severe Handicaps

Final Report of U.S. Dept. of Education Grant No. H180P90015
Submitted to the U.S. Dept. of Education
Division of Technology, Educational Media and Materials

Carrie Brown, PhD
Director, Bioengineering Program
Dept. of Research and Program Services
The Arc
National Headquarters
(formerly the Association for Retarded
Citizens of the United States)

Catherine Wyatt
Technology Coordinator
Bioengineering Program
Dept. of Research & Program Services
The Arc
National Headquarters

Sue Eades
Administrative Secretary
Bioengineering Program
Dept. of Research and Program Services
The Arc
National Headquarters

Maggie Sauer, M.S., CCC-SpL
Assistant Director, Bioengineering Program
Dept. of Reserach and Program Services
The Arc
National Headquarters

Al Cavalier, PhD
Consultant
University of Delaware

TABLE OF CONTENTS

Abstract

Enclosed is the final report for The Sound-to-Speech Translation System Utilizing Graphic Mediation Interface for Students with Severe Handicaps Project, U.S. Dept. of Education Grant No. H180P90015, submitted to the U.S. Dept. of Education, Division of Technology, Educational Media and Materials. The Sound-to-Speech Translation System Utilizing Graphic Mediation Interface for Students with Severe Handicaps (STS/Graphics) system is a voice recognition computer-based system designed to allow individuals with mental retardation and/or severe physical disabilities to communicate with others and to control their environment. This includes individuals with quadriplegia, spinal cord injury, cerebral palsy, aging problems, arthritis, and assorted neurological disorders. Operation of the STS/Graphics System is through vocalization, switch closure, or keyboard input for activation of electrical appliances and/or digitized speech output. Choices available to the user for system operation are presented as customized photographic-quality symbols representing familiar items/persons in the user's environment and are displayed on a monitor. The user makes choices through direct selection, linear scanning, or row/column scanning.

This research project had four major outcomes: 1) development and refinement of the STS/Graphics system; 2) application, evaluation, field testing, and research results of the STS/Graphics system with students having mental retardation; 3) development of training strategies and a training manual on the use of the STS/Graphics system; and, 4) documentation of the STS/Graphics system research with two videotapes: **Voices of the Future** and the **Operation of the STS/Graphics System.**

The STS/Graphics system represents a powerful and versatile means by which children and adults with severe disabilities can increase their independence and productivity. Assessment and educational applications are additional uses of the system. This research project is a major step towards the use of speech recognition as a natural means of communication, learning, training, and computer access by students having mental retardation and other disabilities.

A Comparison of Actual Accomplishments to the Goals Established for the Period

The Sound-to-Speech Translation System Utilizing Graphic Mediation Interface for Students with Severe Handicaps project initially was funded as a 15 month project. A no-cost two month extension was granted making it a 17 month project. Each goal and objective of the project is listed below is specifically addressed. For some Objectives, there is more detailed information available in other Sections of this report.

Goal 1: Outline, modularize, develop and configure the design of the STS/Graphics system software.

Objective 1 *Design the software.*

This objective was completed. See the Software/Hardware Section.

Objective 2 *Program the STS/Graphics system software.*

This objective was completed. The Software/Hardware Section contains photographs of the STS/Graphics system software screens. The attached videotape **Operation of the STS/Graphics System** documents the operation of the system.

Objective 3 *Assemble the STS/Graphics system hardware and development software.*

This objective was completed. The Software/Hardware Section contains photographs with description of the STS/Graphics system hardware.

Objective 4 *Alpha Test the STS/Graphics system.*

This objective was completed in the Bioengineering Program laboratory.

Objective 5 *Document the STS/Graphics system code.*

This objective was completed. The Software/Hardware Section contains examples of the software code which has documentation done by the programmers. The software has been copyrighted with the U.S. Copyright Office.

Goal 2: Conduct the technical and clinical evaluation of the STS/Graphics system.

Objective 6 *Identify Subjects.*

This objective was completed. Specific information on the subjects is found in the Methodology Section. Evaluation documents are in the Evaluation Section.

Objective 7 *Beta Test the STS/Graphics system.*

This objective was completed. Specific information on beta testing is included in the Methodology Section of this report.

Objective 8 *Refine the STS/Graphics system.*

This objective was completed. Refinement was ongoing throughout the project.

Goal 3: Analyze the data and write reports.

Objective 9 *Analyze the data.*

This objective was completed. Analysis of the data is found in the Methodology Section of this report. The publications and reports of this project are listed and contained in the Publication Section.

Objective 10 *Evaluate the STS/Graphics system hardware.*

This objective was completed. Evaluation of the STS/Graphics system was ongoing and specific information can be found in the Software/Hardware Section of this report.

Objective 11 *Develop the Training Manual.*

This objective was completed. The Training Manual can be found in the Training Manual Section of this report.

Goal 4: Disseminate Information

Objective 12 *Develop a documentary videotape.*

This objective was completed. Two videotapes, **Voices of the Future** and **Operation of the STS/Graphics System**, are included with this report. **Voices of the Future** is a documentary of the research process. **Operation of the STS/Graphics System** documents the operation and setup of the STS/Graphics system.

Objective 13 *Disseminate information.*

This objective was completed. The articles which were written and the professional presentations regarding this research are found in the Publication Section. It is the intent of the research to submit the research findings of this research to the journal of **AAC: Augmentative and Alternative Communication** for publication in the immediate future.

A major point of information is that the STS/Graphics system received recognition as one of the top eleven national finalists in the Johns Hopkins National Search for Computing Applications to Assist Persons with Disabilities, 1992. The American Speech-Language-Hearing Association solicited a presentation of this research at their 1992 national convention.

Goal 5: Market the STS/G Software/Hardware

Objective 14 Involve Marketers with the project.

This objective was not completed. Details of the efforts made to involve a manufacturer and reasons for lack of success are as follows:

The Arc approached multiple vendors of either augmentative communication products or Software/Hardware publishers in the assistive technology field regarding the STS/Graphics system. Companies contacted and outcomes were as follows:

Prentke Romich (PR): With travel money from this project, in February of 1991, project staff travelled to Wooster, Ohio, and demonstrated the STS/Graphics system to Prentke Romich staff. Prentke Romich decided that the Graphics interface and the clinical and training applications of the system were extremely powerful. However, since PR develops stand alone units, it did not think that the color graphics approach would work with their products. They had also recently invested heavily in expanding their operation and in developing some new products. They did not have the capital to invest in new products, especially anything that was not in keeping with their product line.

Apple Computer, Inc.: The STS/G system was demonstrated to Mr. Gary Moulton of Apple Computer, Inc. He was very impressed with all components of the system, especially its customizability, speech recognition and training potential. He said that Apple Computer did not have the resources to convert the system from a PC platform to a MacIntosh platform but he helped facilitate extensive discussions with Don Johnston Development Company.

Don Johnston Development Company: The STS/Graphics system was demonstrated to Don Johnston and his staff in October, 1992. The company had just invested heavily in Ke:nx and had just released it to the public. An additional problem was that most of the Don Johnston products are developed on a MacIntosh platform and the STS/G system is on a PC platform. There have been subsequent discussions about obtaining Small Business Innovation Research funds with Don Johnston to change the system from a PC platform to a MacIntosh platform. Nothing further has been done to date.

Johns Hopkins National Search for Computing Applications to Assist Persons with Disabilities: Technology transfer and linking manufacturers/publishers with inventors was a stated outcome of the Johns Hopkins National Search. As a national finalist and winner in this competition, project staff attempted to take advantage of the technology transfer opportunities that were presented. In actuality, there were few opportunities for realistic discussions about technology transfer so this opportunity did not result in the STS/Graphics system coming to market.

Laureate Learning Systems: Mary Wilson of Laureate Learning Systems was impressed with the potential of the STS/Graphics system. However, it was not within the product line or the hardware platform of Laureate Learning Systems products. She connected The Arc with someone interested in technology transfer but it never came to fruition.

products. She connected The Arc with someone interested in technology transfer but it never came to fruition.

Access Unlimited of Houston, Texas: Access Unlimited was not able to support the STS/Graphics system marketing and distribution, but it was willing to offer the service of creating the photographic libraries for buyers of the system.

Baylor Biomedical Services: BBS carefully evaluated the STS/Graphics system. After much deliberation, Baylor decided that it did not have adequate financial reserves to bring the system to market in the immediate future.

Additional funding for this project was received by:

The Arc Research Fund
Martin Marietta Aerospace

METHOD

Rationale

Current research recognizes the power of using familiar items/activities within natural teaching environments/sequences for persons with mental retardation. A focus of this research is to determine the ability of the subjects to make meaningful choices of preferred and familiar items in their environment. After the ability to use voice for selection purposes was established, the researchers extended the application of the STS/Graphics system to novel items and their picture representations. By initially starting with familiar items, researchers were not training the subjects for cognitive and language understanding about the meaning of new objects or photographs. But more importantly, after learning how to operate the system, the subject was immediately able to communicate about an item that s/he knew about and liked. By including items from participants' natural environment on the monitor display, it was felt that their performance and ability to generalize the concepts would be enhanced. For these reasons, the items used with the subjects during the subject selection and evaluation phases were included among the items used during the training and intervention phases of the research project. The STS/Graphics Project operated on these premises in designing, and teaching participants to use, the system.

Additionally, by enabling children to use their own voice as well as their own personal "speech" referents for the items, their ability to operate the system would again be enhanced. The STS/Graphics system was designed to provide enhanced opportunities for independence through communication and environmental control by acting as a vehicle for speech translation and action.

Potential Subject Selection

Subjects were selected from the schools in the Dallas/Fort Worth area. Initial evaluations were made in the Hurst-Euless-Bedford (H-E-B) Independent School District and the Irving Independent School District. H-E-B has 18,768 students in 28 elementary schools, 5 junior high schools and 2 high schools. With 1,909 students in special education, there are 83 students who are identified as having mental retardation, with 48 in the elementary schools, 17 in the junior high schools, and 18 in the high schools. Irving Independent School District has 25,064 students in 17 elementary schools, 9 junior high schools and 3 high schools. With 2,727 students in special education, there are 187 identified as having mental retardation with 126 in elementary school, 30 in junior high school and 31 in high school. The 6 subjects selected for final participation in this research were evenly split 3/3 from each school district.

School administrators were contacted in order to assist the STS/Graphics project team in the identification of potential subjects. Letters were sent to school administration describing the project and subject selection criteria.

School administrators identified teachers in their school systems who could more effectively identify potential subjects. Interviews were conducted with relevant school staff to assist in this determination.

Letters were sent to the parents of potential participants, whose names were derived from the previous activities, explaining that their child was being considered for the STS/Graphics project. This was followed by brief interviews with the parents who responded to the letter to determine whether their child would meet some of the named criteria. If parent descriptions suggested that the child may meet the criteria, parents were asked to sign a release of information form for the school to enable the project staff to review school records. This was necessary to help determine whether or not the child had mental retardation and testing and performance scores were on file as documentation. Other relevant information was recorded such as medical information, etc. which suggested a student's performance and behavior in school, school attendance, and parent support.

Criteria for Subject Selection

Criteria for the selection of the subjects in the STS/Graphics research project included:

- adequate vision and hearing
- adequate positioning and seating
- understanding of picture/object association
- understand cause/effect relationships
- rudimentary scanning skills (if appropriate)
- speech is not functional as a primary mode of communication; i.e., the subject used unintelligible vocalizations
- intellectual and adaptive behavior measures that classified the student as mentally retarded

Final Subject Evaluation

There were two phases to the subject evaluation. The first phase was based on parent/caregiver/teacher interviews. The second phase was based on direct evaluation of the children.

Phase 1: Parent/Caregiver/Teacher Interviews

Following the initial parent interview and record review described above, 24 students (12 male and 12 female) were selected for further evaluation. Potential subjects were evaluated ranging in ages from 4-21 years. Twelve attended school in the H-E-B school district and 12 attended school in the Irving school district. Diagnoses included cerebral palsy, autism, encephalitis, Down's syndrome, and renal dysfunction. Subjects varied level of mental retardation from mild to profound. Of the 24 subjects evaluated, 14 were ambulatory and 10 were non-ambulatory.

As part of this evaluation process, a more detailed interview was conducted with one or both parents/caregivers and school staff. The interviews provided the following information:

1. Medical status with regard to overall health and medication;
2. Sensory skills with regard to vision and hearing skills;
3. Gross and fine motor skills;
4. Seating and positioning considerations;
5. Cognitive skills;
6. Receptive and expressive language skills;
7. Communication strategies;
8. Previous experience with augmentative communication assistive technology;
9. Preferred items in their home and school settings; and,
10. Whether or not specific speech sounds were used consistently to reference these preferred items in their environment.

Questions were conducted with familiar staff, family and peers to identify preferences, behaviors, and vocal patterns in the natural environment. Students were present in the interviews, but the main purpose of the interviews was to obtain information through observation and from others regarding student ability, skills and preferences. This information was recorded on the interview form by the project staff (see the Evaluation Section for the *Interview Form*). It was important that the subjects demonstrated that they had consistent and meaningful speech sounds. It was assumed that this ability would make it more likely that the subject would be able to make effective item selections from the STS/Graphics system. Parents were asked to bring some of the reported preferred items with them to the evaluation so they could be incorporated in the evaluation process.

Research assistants wrote down the students' spontaneous utterances and imitations with their associated item. The interviews, record review, and observations provided the necessary information for the child's participation in the project. The information was used to plan later project activities if the student was selected as a STS/Graphics research subject.

Phase 2: Direct Evaluation of Potential Subjects

Evaluation Procedure to Identify Beta Subjects.
There were certain basic skills needed by any subject to be able to use the STS/Graphics system:

1) *picture/object association*

2) *cause/effect relationships*

3) *rudimentary scanning*

4) *sound production*

5) *definite preferences*

Picture/object association was necessary to insure that the subject understood that the item displayed on the computer monitor in the form of a photographed image represented the actual item. Cause/effect understanding was necessary to ensure that the subject understood that when making a vocal selection from the items presented, that the resulting action was caused by the vocalization. Scanning skills were necessary to ensure that the subject understood how to make a selection using a scanning routine, if direct selection was not possible. Sound production was necessary to ensure that the student could use his/her voice to control the STS/Graphics system. Demonstration of definite preferences ensured that the items made available to the subject were indeed high preference items for the subject so that there was more assurance that subject behavior was not influenced by items disliked by the subject.

Potential subjects were assessed on their abilities to demonstrate each of these skills according to the following procedures:

1) *Picture/object association skills*. There were two steps in the picture/object association evaluation: item (stimuli) selection and photograph preparation.

Item (Stimuli) Selection. With the item/stimuli selection, researchers felt that the ability of the potential subjects to make choices from between two alternatives would be adequate for effectively operating the STS/Graphics system. Choosing between two items was the criteria for success in this part of the evaluation. For information purposes, some students were given the opportunity to make more than two choices to determine if they had this ability. Items were used which had been identified during the interview process and provided from home by the parents. Project staff provided additional items in the event that the reported item was not truly motivating to the child.

The students were given two opportunities to make choices. First, two actual items were presented to the student by the researcher holding the items. In the first presentation, the student was asked to point to the item s/he wanted in an effort to identify the more highly preferred item. In the second presentation, the student was asked to show the researcher with item which was named in an effort to ensure that the student could identify the item. If the student was capable, s/he would point to the preferred item. If unable to point, the student would eyegaze his/her choice. Once a selection had been made, the location of the items were reversed from either left to right or right to the left to ensure that the selection was meaningful.

Photograph Preparation and Item Selection. Prior to the evaluation of a student, project staff asked parents and school staff to loan items to the project staff to photograph. Items were photographed in the Bioengineering lab with the items placed on a white background so that the surrounding background of each item photographed was consistent. Thus, the pictures used during the evaluation process were photographed to ensure clarity and freedom from visual background distractors in the photograph. The actual items were then returned to the school or home. The students were introduced to the photographs of the items utilizing the same procedure described in the Item/Stimuli Selection section above.

On occasion, an intermediate step was used with a subject to assist in his/her ability to understand the selection process. The item would be presented beside the photograph of the item so that the student could make the association of the photograph representing the item. Ultimately, the student had to make the choice without the item present in order to become a research subject.

This same procedure was used to introduce the images of the same items on the STS/Graphics system display monitor. First, with a full screen single image, then using one of the images in a 2X2 display (with 3 blank cells), and then two of the images in a 2X2 display (with 2 blank cells).

2) *Cause/Effect Skills*. The students were given an opportunity to demonstrate their understanding of cause/effect by using a voice switch in several activities. All students were asked to use the voice switch regardless of physical ability. It was determined that they had cause/effect understanding when they independently operated the switch upon command from the parent or researcher (i.e., "make it go", "make it stop") and then demonstrated control over this operation without any direction. Evaluation activities includes for use of the computer with cause/effect software programs; small appliances adapted for switches such as tape recorders; and, switch adapted toys.

Cause and effect was evaluated using the voice switch both to play the *R. J. Cooper Early and Advanced Switch Games* on an Apple IIe computer and to turn a tape player on and off. Software games used included Noises, Numbers, and Building Shapes. Using an AbleNet Control Unit, a tape player for music feedback to the student was connected to the voice switch. Several times the researcher demonstrated how the tape player was turned on by producing a sound. After repeated demonstrations, the researcher asked the student to *"Turn on the music"*. Once the student turned on the tape player, the time-out function of the AbleNet Control Unit turned the music off after a short time. The process was then repeated. Success was measured by the subject successfully playing the games or operating the tape player.

3) *Rudimentary Scanning Skills*. The method by which a student would control the STS/Graphics system, either scanning or direct selection, was decided by assessing the students' ability to produce multiple sounds which were spontaneously associated with specific items. Students who did not demonstrate this ability were evaluated for scanning access. Students who did demonstrate this ability were assigned to direct (voice) selection.

For the children who required scanning as a means of control, scanning was evaluated after picture/object association and cause/effect evaluations were completed. The Apple computer was utilized for the scanning evaluation with a software program which closely resembled the "blank display" pages from the STS/Graphics system. Scanning was evaluated by using the Scan and Speak Software on an Apple IIe computer. A single photograph was attached to the computer monitor and the scanning software was activated. The voice switch was used by the subject to halt the scan box. A single photograph of an item previously used during the evaluation was placed on the monitor display. A 2 x 2 display was used (with 3 blank cells). The photograph was placed on the monitor in either of the top two cells. The student was then asked to use the voice switch to stop the "box" on the picture. The photograph was then placed at other cell location on the display. Once a student demonstrated the ability to stop the scan box over a single picture, the same process was followed using two pictures on the 2 X 2 display. The student needed to successfully place the box on the picture five times.

4) *Sound Production Skills*. The preliminary evaluation of the students' ability for sound production was done in the initial interview. Research assistants wrote down the students' spontaneous utterances and imitations matched with their associated item. During this round of evaluation, students were first presented with an object and asked what it was. Project researchers observed the student's response to the presentation of the preferred item and noted the student's use of speech sounds to reference the items, as well as whether or not the speech sound was used consistently with each presentation of an item. The sound that the student produced was written down by a research assistant. This was repeated for several objects. The objects were presented again to verify that the sounds were used consistently with the specific object.

5) *Definite Preferences*. A list of items identified by caregivers and teachers to be of interest to the subjects during the interview process was compiled for each subject. A variety of the objects on these lists were collected and photographed. The photographs were used in the first five intervention sessions (described below) with a subject. It was determined at this time that it was important to rank the photographs according to the subject's order of preference. This prevented any failure by the subjects to use the system effectively due to lack of interest in the items presented because they were not preferred. After these five intervention sessions, intervention was interrupted to perform forced choices for the preferred images. In the forced choice procedure, the subjects were shown photographs of items that were assumed to be high preference items. These photographs were randomly placed in sets of two for presentation to the subject. The list of pairs was constructed such that no photograph appeared more times than any other in either the right or left position and no photograph would appear in the same location twice consecutively. Every photograph appeared with every other photograph one time. The subject was asked to indicate using their personal style which of the two photographs was his/her preference. After a choice, the subject was allowed to interact with the actual item for a few minutes. Forced choice was conducted for approximately 5 sessions for each subject with no intervention during that time. This interruption of interventions occurred for all subjects.

Evaluation Outcomes

In summary, students who were selected for participation in the STS/Graphics research had to demonstrate their ability to understand cause and effect, to perform sound production/imitation, to make picture/object association, and to indicate an ability to scan. Each subject used the STS/Graphics System in her school environment.

Alpha Subjects

Two subjects, who were severely physically impaired but not mentally retarded (and therefore did not meet criteria), were selected as alpha subjects. The alpha subjects participated in a variety of informal tests of the STS/Graphics system to provide feedback to researchers about the soundness of its design and operation.

Beta Subjects

Six students were selected for participation as beta subjects: four females and two males; chronological age ranges 7-21 years; diagnoses of cerebral palsy, encephalitis, mild to profound mental retardation; four were ambulatory (two walked with assistance of a walker), and two were non-ambulatory. The mode for accessing the computer were direct selection for three subjects and scanning for three subjects.

Before the project ended, three subjects remained in the study. Two were direct selectors and one was a scanner. Two of the original subjects were dropped from participation because of health problems. The third subject was dropped from participation because from the time of evaluation until training/intervention, her teacher began to use Touch Window technology with her in the classroom. Consequently, this subject learned to use her hands rather than her voice in operating the computer, even though our assessments indicated voice operation would be a more preferred access mode.

Subject TL was a female from A.S. Johnston Elementary School in the Irving Independent School District. She was 11 years 4 months at the time of her first observational evaluation. Subject TL had severe mental retardation and a seizure disorder which was most predominant when the subject was tired. Medication controlled the number of both grand and petite mall seizures. Subject TL exhibited a fine motor tremor which was became more pronounced when she was nervous, upset or under stress. She was ambulatory.

School records showed that at 10 years 4 months, Subject TL was evaluated and exhibited an age equivalency of 36 months. When scored on both the Stanford Binet and Vineland Scale, her score was 35 or below. Standard scores on the Woodcock-Johnson Psychoeducational Battery fell below 65 and receptive language skills fell within the 3 to 6 year level. An Audiometric Screening was attempted but was not completed. Parents and teachers indicated that her hearing and vision was within normal range.

Communication. Subject TL used speech, signing and pointing to objects to communicate and to indicate her needs. Her mastery of sign language was inconsistent, and signing was becoming less functional as she was attempted to communicate using speech even though it had limited intelligibility. Subject TL utilized one and two word sequences which could be understood by parents and teachers. In the community, (restaurants and the mall) she was learning to use a communication system made of black and white photographs.

Computer Experience. Subject TL used a Tandy computer at home and an Apple computer at school which she controlled by normal keyboard access. The school was utilizing computer games to teach Subject TL turn taking skills.

Subject MP was a female from A.S. Johnston Elementary School in the Irving Independent School District. She was 11 years 7 months of age at the time she was evaluated for participation in this project. She had mild cerebral palsy and seizures which were controlled by medication. According to her school records, Subject MP was evaluated 9 months prior to this project and scored 27 on the Vineland Social Maturity Scale. Subject MP was ambulatory and indicated self help skills at approximately the 48 month level.

Communication. Subject MP was non-vocal but produced a variety of unique sounds. She used a communication board for ordering food at McDonald's and could sign approximately 70 to 100 signs which she used spontaneously and upon request. When asked to indicate her desires, she would hesitate and then walk over to the item. If selection was not possible using this method, she would go through a list to indicate her choice.

Computer Experience. School records indicated that Subject MP had been working on the computer even though no details were recorded.

Subject DL was a female from The Career Center in the Hurst-Euless-Bedford Independent School district. The Career Center is a self-contained school for high school students for transitioning students from high school to employment. At the time she was first observed, Subject DL was 21 years 4 months of age. She had cerebral palsy, fibromuscular Hyperplasia, vascular system constriction in the bilateral renal artery, hypertension and had experienced 5 strokes since age 18. Subject DL had balance problems and left side involvement. Additionally, she had seizures during sleep, mood swings and decreased endurance. Subject DL exhibited limited range of motion and motor skills. Even though she was ambulatory she utilized a wheel chair during exercise or for mobility over longer distances. She did not propel herself, but needed to be pushed.

One month prior to the observation of Subject DL, she was assessed using the Leiter International Performance Scale where she indicated an IQ of 35 and an age equivalency of 2 years 1 month. The Wechsler Adult Intelligence Scale was administered but no results were recorded.

Communication. Subject DL was non-verbal with severe expressive language delays. She used a clip board communication system with line drawings of daily requirements. She also used approximately 75 signs from spontaneous sign language that her parents attribute to her watching a television show that taught signing.

Computer Experience. Subject DL had experience using an Apple IIG and Apple IIE at school with basic academic programs. She had inconsistent success with the computer. During the STS/Graphics project, Subject DL was given an Apple IIE. By projects' end, she had not started to actively use the Apple computer at home.

Subjects Selected to Use the STS/Graphics System

The method used to introduce the STS/Graphics system to the two subjects who controlled the system through direct selection (Subjects TL and MP) was different than the method used with the scanning Subject DL. They will be described separately.

Subject TL was an 11 year old ambulatory female, with severe mental retardation, who used direct selection to control the system. The subject made few vocalizations, all of which were unintelligible. Subject TL held the clip microphone close to her own mouth because she did not like it attached to her clothing. Once she had produced a sound, she would usually hand the microphone to the researcher to free her hands in order to interact with what she had requested.

Subject Descriptions of Scanners. Subject 3 was a 21 year old semi-ambulatory female, with mental retardation, who needed extensive support, and who used scanning to control the system.

Procedure

Instrumentation

Key features of the STS/Graphics System were as follows: (a) photographic-quality computer-generated images of items/persons in the subject's natural environment that were presented on a display to represent choices among spoken communications and environmental control activations; (b) vocalization access, whether intelligible or not, that were used to select choices from the image display; (c) physical switch activations and traditional keyboard inputs that were alternate access modes; (d) direct selection via voice and keyboard access modes; (e) linear scanning and row/column scanning via voice and switch input; (f) speech output that was in the form of digitized, easily-understood speech that was age and sex appropriate for the subject; and, (g) data on system use that was automatically recorded, thereby facilitating the development of appropriate training strategies. A more detailed description of the STS/Graphics System follows:

The following list of hardware components represents the hardware in the STS/Graphics System.

- International Business Machine (IBM) Personal Computer (PC) Advanced Technology (AT) or compatible with a 286 central processing unit (cpu)
- 30 Megabyte (MB) of hard disk space
- Super Vector Graphics Adaptor (VGA) monitor
- Paradise Professional Video Adaptor
- Votan 2000 speech recognition board
- Image capturing device with color capability
- X-10 Powerhouse and various wall modules
- 640 Kilobyte (KB) of RAM
- Floppy drive
- Microsoft-Disk Operating System, version 3.30 (MS-DOS) or greater

(The Software/Hardware Section gives detailed information about the STS/Graphics system hardware, software, training of voice templates, environmental control, sound-to-speech translation and voice template matching, and access).

Microphone Issues

During the initial stages of field testing, difficulty was experienced in locating the microphone in a position that was not a distraction to the subjects and yet allowed utterances to be "recognized" by the system. Reaction to the microphone varied from subject to subject, but in all cases it was a distraction from the items displayed on the monitor. It was decided to use one of two microphones with the students, either the stand alone Votan microphone/ speaker unit, or a Realistic clip-on microphone, positioned in different locations for different subjects.

Access Modes

The STS/Graphics software provided two methods of accessing the system: direct selection and scanning. In the direct selection mode, a subject vocalized a different sound for each image that was presented on the display. The scanning selection mode was operated in two ways: linear scanning or row/column scanning. In either scan mode, the system required only a single vocalization to access all choices and the vocalization did not have to be consistently pronounced from one occurrence to the next. The vocalization served to stop the scanning sequence. Once the scan halts, the functions executed that were associated with the image on which the scan stopped (that is, speech output, environmental control, or both). Alternately, the scanning could be stopped by using a single switch.

Voice Input for Training Speech Templates

When operating the STS/Graphics system, the subject's voice functioned as a switch and the subject "trained" the computer to recognize by speaking the target vocalization into the microphone. This sound was recorded digitally on the hard disk. The training was repeated three times and each vocalization was recorded separately. Once the three trainings were stored in a subject-specific voice template, the STS/Graphics software analyzed the three recordings and averaged them.

Creating Voice Templates for Direct Selectors

To create voice templates for direct selectors (this software function has been redesigned), a photograph of the desired object was held close to the microphone, the student was prompted to speak. The direct selection prompting sequence was:

1. *"Tell me about the ______"*, or
2. *"Use your voice to tell me about ______"*, or
3. *"Tell me about the ____"* - while pointing and taping the appropriate object on the monitor, or
4. Modelling of the sound by the researcher

The system's record function was activated, and the utterance produced by the student was recorded by the system. This process was done twice and then tested by repeating the process again with the voice matching function activated. If the third test utterance was within an acceptable range of the two recorded utterances, then the recordings were saved into the voice template. If the matching range was inadequate, then the training process was repeated until the match was acceptable. This process was done without the computer activating any speech output or environmental control action in response to the vocalizations during the voice training process.

Creating Voice Templates for Scanners

For scanning access to the system, the subject specific template was comprised of trainings for one unique utterance per subject. The scanners prompting sequence was:

1. *"Show me the ______"*

2. *"Show me the ______"* - tap monitor in the correct cell when scan box is on that cell.

3. The researcher would demonstrate how to stop the scan action by vocalizing at the correct time and halting the scan action over the displayed picture.

The scan box started moving through the 2 X 2 matrix, the researcher pointed to the scan box and whispered *"not yet"* or *"no"*, and sometimes shook her/his head "no". This procedure was repeated on each empty cell. When the scan box was on the picture, the researcher tapped the screen more actively and whispered *"yes"*. Praise, clapping, smiles etc., reinforced each successful activation.

Once all of the trainings were stored, the system checked to verify that the subject's target vocalization would successfully match one of the sounds stored in the voice template. If the match was not similar enough to the original trainings, then the subject would retrain the voice template for that particular utterance.

When the subject was actually operating the system and producing a vocalization, the system checked to see if the vocalization achieves a "match" with the trained vocalizations stored in the voice template. In the event that there was a match, the system would acknowledge the vocalization by activating the appropriate speech output message or environmental control function; in the event of the vocalization not matching any of the trainings closely enough, the system would not respond. If the system did not respond, two optional corrective actions could be taken. First, the accuracy used by the system to match a vocalization to the stored voice trainings could be made less sensitive so that the system was less demanding in its matching requirements. Second, the subject could choose to repeat the training process to refine a vocalization stored in the template for better recognition accuracy.

Sound-to-Speech Translation and Expansion

For communication purposes, each subject could access speech output messages stored in digitized files on the hard disk. Each message could consist of speech of up to 8 seconds in duration, thus permitting single words or complete sentences to be output. A model whose voice was age and sex appropriate for the subject recorded these messages into the system. The subject's often brief and unintelligible vocalization translated into intelligible speech and expanded into a phrase or complete sentence. For example, if a subject's vocalization for *"water"* approximated *"wuh"*, the system could immediately output *"Could I have a drink of water, please."* Message files for many different subjects could be stored and accessed via one computer system.

Environmental Control

Any voice (or other) input could be linked to the activation and deactivation of designated electrical devices. A single voice input could also be linked to any combination of speech outputs and device activations, e.g., the vocalization *"pa"* translated into *"I'm going to watch TV now"*, and the television set turned on.

Graphics Interface

The system displayed on the video monitor photographic-quality images of objects, appliances, and people in the subject's environment. The photographs with the highest ranking items were scanned into digitized images so they could be presented on the STS/Graphics display. These images were associated with spoken communications, environmental control, or a combination of both. While all of the subjects in the research used a 2 X 2 matrix of images, the present system's capability allowed the size of the images, as well as the number of images appearing on the display, to be customized to the subject. Displays ranged from a single image to a 5 x 5 matrix of images. The graphics display could also "flip" to a second page of choices for each subject. The two pages of images permitted the configuration of a hierarchical system of choices for subjects who understand the concept of categorical clustering.

Research Design

Researchers investigated whether subjects with mental retardation and/or multiple disabilities could learn the cause/effect relationship between vocalization and function. If subjects learned the cause/effect relationship, could they then learn that different vocalizations are associated with different functions? Would the subjects exhibit increased communication with others as a result of system use? The viability of the graphics component as an option for this population using this type of communication/environmental control system was also an important issue. And finally, what aspects of the system, if any, needed further refinement?

Experimental Sessions

Phase One: Training of the Subject on System Operation with a 1 X 1 matrix

In the first phase of the research design, training on system operations was done using a photograph of the object; later the image was displayed on the monitor. Training strategies were taught to the subjects on how to operate the system using their personal vocalizations. During single-image trials, there was only one graphics image displayed on the monitor in a 1 X 1 matrix (meaning a single image filled the screen), and the corresponding speech output or device was activated.

Phase Two: Introducing a 2 X 2 matrix with one image.

In the second phase of the research design, additional graphics images were added to the display in a 2 X 2 matrix and their corresponding choices were available for activation. Subjects would advance from single-image displays to multiple-image displays when they successfully completed at least nine out of 12 trials on each type of display.

In both phase one and phase two, experimental trials consisted of the researcher orally requesting the subject to activate a specific device or communication phrase, e.g., *"Tell me about the beads"*. A trial was scored as successful when the subject emitted the vocalization that corresponded with the requested device or communication and the computer system recognized it as such and activated the proper output. Three levels of prompting could be used with subjects to help them attend to the monitor display and to learn either the direct selection or scanning strategies. These prompts consisted of verbal instruction, tapping on the display, and/or modeling the subject's vocalization to aid the subject in learning to correctly vocalize and to control the system.

Phase Three: Introducing 2 x 2 Matrix with Two Images

Explanation to the Subject. The subject was informed at the beginning of the session that *"We will be doing something a little different/new today. I want you to tell me about/find the pictures. Later, you can ask/show me the thing you want to do."*

These expectations were established early in the 2 x 2 display level of training.

Selection of Items for Multi-item Display. Items were be included on this display which had the "strongest preference value" in the subjects' hierarchy. Since these items will be used continuously for the duration of the subject's participation in the study, careful consideration must be given to the selection of these items. Selection should be based on subject's original forced choice selection data as well as overall performance with the items during the 1 x 1 and 2 x 2 single items array.

Selection of images included on the 2x2 display. Images available from the subjects' original "forced choice" lists were used to make decisions regarding the introduction of multiple images on the display. Those items which had the "strongest preference value" as demonstrated by the forced choice selection procedures conducted for each subject, were used to create the multiple item displays. For the purposes of the 2 x 2 display, four images were selected which were topically similar (i.e., beauty aids, leisure activities, snacks) and demonstrated the "strongest preference value". The two items demonstrating the "strongest preference value" were introduced on the 2 x 2 display first. In some cases, the item of strongest preference "overpowers" all other choices presented on the display. In this instance, the item should be introduced as the fourth item on the 2 x 2 display.

The rationale for using this type of image selection for the 2 x 2 or other larger displays introduced later in the study, is based on the amount of time the subject will "experience" the item. Should the subject complete criterion successfully at each level of 2 x 2 display activities, the subject would be introduced to the next display size (i.e., 3 x 3). It is imperative that when the items used during the 2 x 2 display level of testing are incorporated on the larger display, that they retain their "preference value" throughout present and subsequential training phases.

Definitions. In order to facilitate the reader's understanding of the following procedures, the following definitions are given:

1) *Set*: since the STS/Graphics system equipment required time to regenerate each new display screen, subjects were given consecutive three trials with each display format, i.e., with the same images displayed in the same cells. This group of three trials is defined as a "set". The display was changed each time a "set" was presented to the subject.

a. Experienced set: a set of trials to which the subject has previously experienced. For this portion of the study, sets contained within block two and three (defined below) are "experienced sets".

b. Novel set: a set of trials which used display formats the subject has never seen i.e., the same images displayed in different cell locations.

2) *Block*: a collection of sets. The order and configuration of the displays used in the block was pre determined. There are six possible display formats. The subject experienced only four of the six display formats.

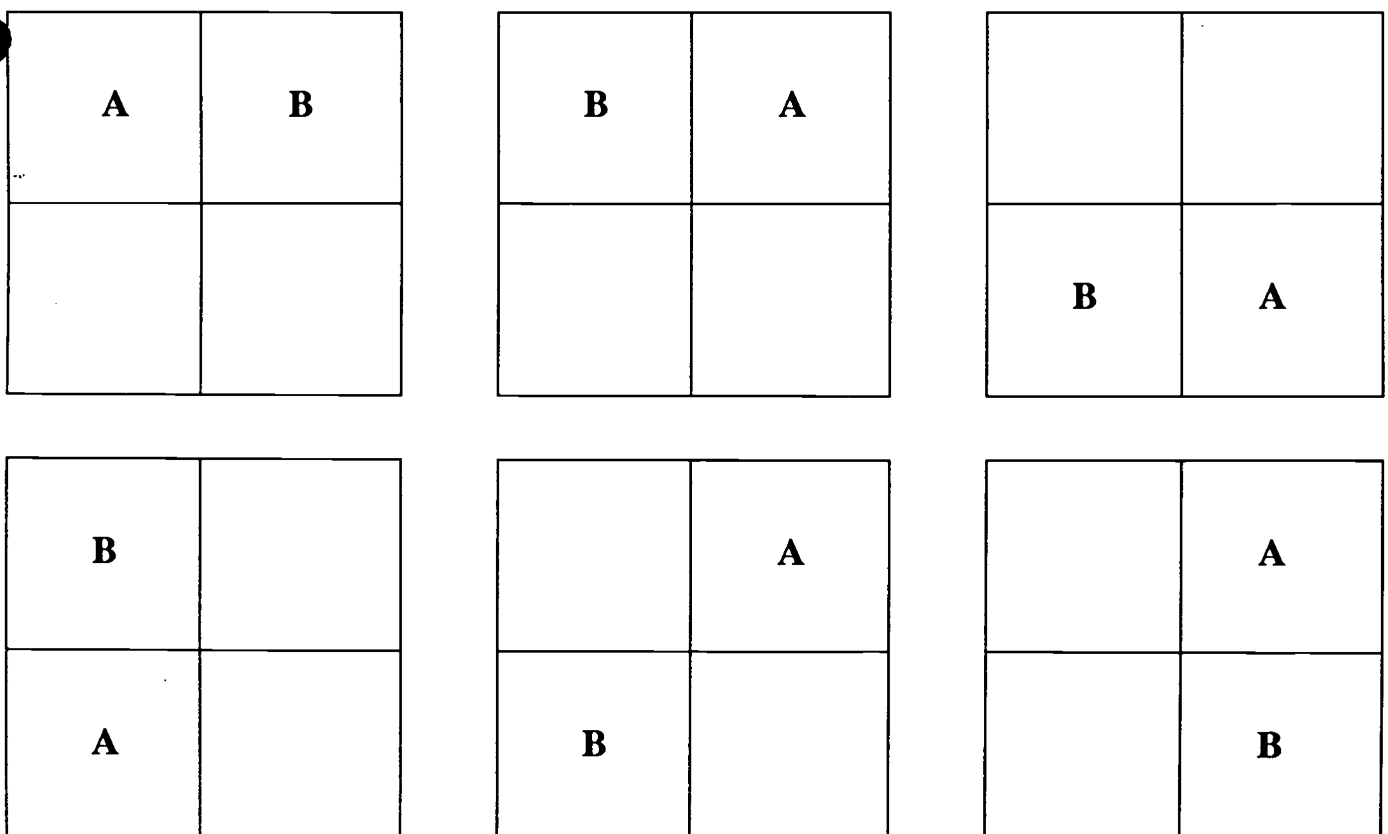
a. *Experienced Block* a block to which the subject has previously experienced.

b. *Novel Block:* trials which used display formats the subject has never seen i.e., the same images displayed in different cell locations.

c. *Generalization Block*: This block contains a total of six sets of trials. Four of the six sets are "experienced sets." In other words, the subject has had the opportunity to experience and use these sets previously. The remaining two sets are "novel sets" which had never been presented previously to the subject. For this portion of the project, the generalization block was presented as part of the third (final) block administered to the subject. The block was composed of four experience sets and two novel sets.

Display Format

1) *Display Construction* -- Each display was created using two photographic images. Each image was placed on the display as designated below:



Displays FIVE and SIX will be designated as "generalization displays" and will not be administered prior to successful completion of BLOCK THREE! Presentation of displays should follow order.

Criterion

1) *Block One* -- Each subject was provided with the opportunity to demonstrate their control of the STS/Graphics System. The first test block was comprised of four different displays. The subject had to obtain 80% accuracy, or nine out of 12 trials for each item.

If criterion was achieved, the subject was moved to the next level, a progression to a 2 x 2 matrix with three images.

2) *Blocks Two and Three* -- If criterion is not achieved, the subject will be exposed to blocks two and three. A criterion of 80% had to be achieved in block three in order for the subject to continue in the study. If 80% criterion was achieved, the third block of trials included the standard block three in addition to the generalized blocks.

Presentation Procedure. Operational Competence

Time: 30 minutes

1) *Presentation order* -- The order which images are presented to the subject was determined by a list containing random ordering of possible two image pairings. The total trials for each block of trials (i.e., 1, 2 and 3) was 24 with the exception of the generalization block for block 3 (an additional 12 trials was added at the successful completion of block 3). The specified order of presentation was randomized for use with each subject.

a. Prior to the use of any prompting sequences, the initial session should be completed using only statement 1.a. below. It was videotaped and maintained as baseline information.

2) *Videotaping* -- For this portion of the training session, the videotape footage captured the affective behavior of the subject and the researcher or interactant. The camera was moved from the subject profile shot, which included the display, to capture the interactions (faces) of the subject and those interacting with them on the system.

3) *Procedure* -- The subject was provided with a two-minute waiting period before presenting the next level prompt.

a. The subject was told , *"Now you can use the computer to ask for what you want."*

b. If there was no response to this statement, the researcher should ask, *"Do you want some ______ or ______?".*

If a successful activation occurs, then return to 3.a. Prompt the subject if necessary.

c. If no response to the researchers statement in 3.b., the research assistant then said, *"How would you tell me you want the ______ or ______?".*

If a successful activation occurs, then return to 3.a. Prompt the subject if necessary.

d. If there was no response to statement 3.c., the researcher said, *"How would you tell me you want ______?".* This was used with one item and used with the second item if there is no response.

If a successful activation occurs, then return to 3.a. Prompt the subject if necessary.

e. If there was no response to statement 3.d., the researchers modeled the communicative behavior using the subject's display. Researcher one used the system using the subject's target sound or scanning motion to communicate the preferred item to Researcher 2 as follows:

If a successful activation occurs, then return to 3.a. Prompt the subject if necessary.

Researcher 1: *"I'm going to try it."* The researcher activated system using the subject's direct select sound or scanning motion.

Researcher 2: Acknowledged the request by providing the activity or item. In addition s/he stated, *"Here is the ______."*

f. The subject was prompted using the statement from 3.a. If there was no response, the researcher said, *"Let's try ______ together."*

If a successful activation occurs, then return to 3.a. Prompt the subject if necessary.

g. If there is no response by the subject to statement 3.f., the activity was discontinued and noted on the data sheet.

Scanning Access

The subject was positioned in front of the monitor/microphone, the picture was displayed, and the subject was asked to *"tell me about the ______".* Whatever sound was vocalized was recorded by the computer. Only one sound was recorded and it became the switch to stop the scanning process.

It was not necessary for the subject to produce a specific sound initially because the STS/Graphics system was used with a very "wide" recognition range, therefore it would activate with any sound the subject made. As the subjects became more familiar with the system, they produced a more consistent sound and so that the recognition range of the speech recognizer was narrowed. During the initial phase, the room was very quiet while the computer was active since background noise could activate the system.

Corrective Procedures for System Malfunctioning

Within each of the phases described above, systems occasionally occurred. When these malfunctions occurred, it was necessary to do a check of the system for the integrity of the voice recognition function. When those checks were necessary, the following procedure was observed. Each subject was seated in front of the monitor and microphone and the full screen display of an image was shown on the monitor. The researcher prompted the subject *"tell me about the ____"* then pressed ENTER. The next sound that the computer "heard" was recorded by the computer and the machine was ready to repeat the sequence. After two trials of these recordings, the software sequenced automatically to the test mode.

A test step was introduced to test the sounds of direct selectors because of ongoing hardware problems with multiple sounds. These problems occurred when a third unique sound was introduced into the voice template for a subject. In the test mode, the subject was prompted to repeat a sound, the machine recorded it and then compared the sound to the two prior trainings. Depending upon the quality of the match, the monitor displayed a statement identifying a "Good recognition" or that the software was "Unable to recognize" and the researcher decided what step to take next.

The next steps options available to the researcher were:

1. If the training was recognized as adequate, then another sound was recorded.

2. If the training was recognized as adequate, the subject used the sound to activate pages displaying the image related to that sound in the STS module of the software.

3. If the training was not adequate, then the entire training sequence for that sound can be repeated

After the third sound had been recorded the reliability of the system to recognize and match each of the three sounds to their assigned pictures was degraded. The degradation was inconsistent from day to day and therefore once sounds were trained in the manner described above, a display page (specifically configured for test purposes) was displayed on the computer monitor by the researcher. The subject was requested to *"tell me about ____"* for each of the three objects displayed. If the sounds activated the prompted items displayed on the page, then field testing continued in a standard manner; if the sounds did not activate correctly, then they would be retrained.

Field Testing

In the middle of August 1990, phase one of the STS/Graphics system was introduced to the six subjects. This provided a full screen display of an image.

Intervention started in July 1990 with all six subjects. During initial sessions, one research assistant videotaped while another took notes on activities during a session and also worked with the subjects with the researcher.

The computer system was setup in a room approximately 40 X 20 that was isolated from other rooms in the school for training purposes. Furniture was moved to provide the least distracting environment and best access to the system. Objects to be used for a specific session were placed in a box near the researcher or the research assistant.

The subject entered the room, greetings were then exchanged, and the subject was introduced to everyone until she became comfortable with the research team. The subject was positioned in front of the system.

The procedure for preparing and working with each subject varied slightly so it will be briefly described for each one.

Subject TL - At the beginning of a session, Subject TL would be very active, examine everything in the room, especially the video camera. Once finished, she would sit in front of the monitor. Her chair would need to be pushed close to the table to keep her attending and sitting. Sometimes she would wear the microphone and on other occasions she would choose to hold it.

Later in the project, to assist her with attention, Subject TL used a head-set instead of the external speaker for the STS/Graphics system. In a few of the last sessions, she sat on the researcher's lap because she was very active and did not want to remain sitting.

During sessions with particular research assistants, Subject TL would work for about 20 minutes. Each time she successfully activated the system, she would "High Five" one assistant and then turn to interact with the other and any other people in the room.

Subject TL went through the steps of the research design as defined. She had problems with producing consistent L sounds that the computer could discriminate when "lip gloss" and "lotion" were introduced. She used *"li"* for "lip gloss" and *"lo"* for "lotion" but they were too similar for the computer to discriminate.

Subject MP - Subject MP normally arrived with a "goodie" bag which was examined by a research assistant to see which preference items Subject MP had brought from home. Items that were to be used as preferred items would be but in the box with others and the bag would be put out of sight. Subject MP was initially resistant to wearing the microphone, but over an extended period of time, she began to wear it most of the time. Occasionally she would choose to hold it.

She was nervous and apprehensive at times and normally answered *"no"* to questions and initially said *"no"* to the computer but quite quickly settled down and used her sounds. When she decided a session was finished, she went back to *"no"* and when requested to use her sounds, would say *"no"*.

Subject MP stepped through the research design.

Subject DL - Subject DL arrived with a bag, containing her school schedule and communication book, which she gave to a researcher. She greeted everyone and then would be happy to sit in front of the monitor and would get to work. She always wanted to scoot her chair forward and would need help. Subject DL worked both with and without the light.

Subject DL worked through the research design as described in the data analysis sheets. She advanced to a 2X2 display with a single image.

Routine

Once the subject was seated and comfortably positioned the first display was loaded on the computer. The researcher would use prompt 1, if there was no response or it was incorrect then the next prompt in the sequence was used, etc. Once the system was activated by the correct utterance, the activity or object was provided to the subject. If the subject was working in the dark, then as soon as the computer reacted to her utterance the light was turned on. The activity was pursued for a few moments, e.g., one page of a book read, one bead slotted, one bit of food. For X-10 items they timed out at several seconds. Later it was found that the research assistants were manually controlling these activities so the timing depended on human judgement.

When the computer was activated, the subject not only received the requested object/item but also was praised by the researcher. Often the subjects would interact with other members of the team to receive affirmations. In cases where a subject showed a reluctance to terminate an activity after the allotted time, another research assistant would ask to see the book or activity for a while. This strategy seemed to be successful. Once the activity was requested again through the computer, the research assistant would give it back to the subject thanking them for letting them see/share the activity. In the case of books, this process was used but the team member asked to see the pictures at the end of each page.

This process was repeated throughout the session. Sessions were normally ended because of time limits.

Results

Subject TL, who used direct selection, progressed through training that began with selecting a single item on the display and ended with selecting from multiple items that were topically similar, e.g., beauty aids, food items. Subject 2, who used scanning, did not progress past a single image on a 2 X 2 display at the time data collection terminated.

The two subjects learned to successfully communicate and to make choices via the STS/Graphics System. They successfully learned the cause/effect relationship between vocalization and action. Subject TL, the subject using the direct selection access mode, learned that different vocalizations were associated with different functions. Variability existed between the subjects in their level of proficiency in operating the system. Subject TL demonstrated an understanding of the use of the system with two choices available, both of which were highly motivating. Her performance deteriorated upon introduction of a third choice. It is unknown if this was due to boredom, dissatisfaction with the choices available, inability to differentiate the target vocalizations with three choices, or a lack of understanding of the use of the system with three choices. Subject 2 demonstrated understanding of use of the scanning access mode to make choices. Although she had multiple items from which to choose, due to lack of time with the system, she never advanced beyond a single item on the display at one time. The graphics component of the system was appropriately used by the subjects in this application.

Aspects of the STS/Graphics System requiring further refinement are (a) facilitating the additional training of the system that sometimes becomes necessary due to the subjects' improved vocalization over time, (b) improving the accuracy of the system's voice recognition in noisy environments, (c) increasing the flexibility of the microphone placement, and (d) improving the portability of the system.

Conclusions

The STS/Graphics System is a viable method of communication and environmental control for children and adults having mental retardation and/or physical disabilities. The system is effective in teaching persons with mental retardation to use their natural speech to communicate and to control their environment, and thereby to become more independent. The graphics interface provided a symbolic visual aid that served to highlight the options available to subjects and facilitate the act of making a selection. Vocalization is a natural means by which to empower an individual and proved to be a robust access mode for assistive technology with this population. Additional applications of the STS/Graphics System for assessment and educational uses will be explored. Given the enhancement in personal choice making that the STS/Graphics System provides its subjects and its potential to make a contribution in other functional areas, future research on the application of voice recognition technology to the needs of children and adults with mental retardation should be revealing.

The efficacy of the STS/Graphics System with each subject was evaluated with a changing conditions single-subject experimental design (Alberto & Trouknan, 1990; Smith, 1979).

Subject DL

Subject DL, the student who used scanning as her access mode, achieved the criterion level of performance under Condition A, a 1 x 1 matrix (a full-screen single-symbol display) using each of eight different symbols, by the ninth experimental session. (See Figure 1.) During Session 8, Subject DL was exposed to both a 1 X 1 matrix and a 2 X 2 matrix displaying a single symbol. Her performance during these nine sessions was highly variable, ranging from a low of 0% of correct responses to a high of 100% of correct responses. Her mean performance during Condition A was 58%.

Subject DL's performance under Condition B, a 2 x 2 matrix displaying a single symbol (and three blank cells), maintained this high variability through Session 17. Thereafter, Subject DL's performance showed much greater stability, ranging from a low of 47% to a high of 77% of correct responses. Her mean Performance during Condition B was 59%.

The cognitive demands on Subject DL were markedly different under Condition A than under Condition B. The task under Condition A required Subject DL to vocalize while a single symbol was visible on the display. The vocal response could vary within a relatively wide range and still be recognized by the STS/Graphics System. Also, the same vocalization was acceptable for any of the eight symbols displayed. The only requirements placed on Subject DL were that the vocalization exceed the voice-recognition threshold of the system and that it be spoken within a reasonable latency after the initiating prompt from the experimenter. That is, since in the scanning mode a subject's vocalization merely acts as a "voice switch", the exact nature of the vocalization is less important than its timing. And in the case of a 1 x 1 display, there is essentially no scan (i.e., there is only one symbol centered on the display) and consequently the timing demands are also minimal. The "correctness" of the response was indicated to Subject DL by the resultant activation of speech output or an electrical device.

On the other hand, the task under Condition B on any given trial displayed a symbol in any of the four cells in a 2 x 2 matrix. Subject DL was required to recognize that this divergent visual display served the same function as the previous, simpler visual display. To lean that the colored "box" that surrounds a particular cell and that continually moves from cell to cell is a discriminative stimulus for vocalizing; and to anticipate the movement of the box to the cell containing the symbol. The critical cue was actually a discriminative-stimulus complex composed of the symbol and the box. When these two stimuli were combined, they were discriminative for the vocal response. The "correctness" of the response, i.e., a recognizable sound that was vocalized while the box was surrounding the symbol, was indicated to Subject DL by the resultant activation of speech output or an electrical device.

Despite the increased visual complexity and task demands under Condition B, Subject DL's performance did not deteriorate compared to that under Condition A. This suggests that Subject DL came to understand the general utility of the system as an outlet for communicative intent and to learn the more demanding operational requirements of the system. In fact, the increased stability of Subject DL's responding across the last 19 experimental sessions suggests that her learning continued to strengthen and her use of the system became more reliable as her experience with the system increased. Based on this evidence, we have every reason to believe that Subject DL would continue to refine her use of the system with a 2 x 2 matrix with a single symbol and transfer quite readily this learning to a 2 x 2 matrix with multiple symbols.

Subject TL

Subject TL used direct selection as her access mode with the STS/Graphics System. Under Condition A_1, a 1 x 1 matrix (a full-screen, single-symbol display) using each of three different symbols, her performance increased from 25% correct responses during the first experimental session to nearly 94% correct responses during the second session. (See Figure 2.)

Under Condition B_1, a 2 x 2 matrix displaying a single symbol (and three blank cells), Subject TL averaged 72% correct responses with two of the symbols used in Condition A_1. Under Condition A_2, a return to a 1 x 1 matrix, Subject TL achieved a criterion level of performance with all six symbols by the next three sessions (Sessions 7-9). During Session 9, Subject TL was exposed to both a 1 X 1 matrix and a 2 X 2 matrix displaying a single image.

Under Condition B_2, a return to a 2 x 2 matrix displaying a single symbol, Subject TL's performance ranged from a low of 27% to a high of 95% correct responding. Her mean performance was 67% for the 10 experimental sessions (Sessions 10-19). Under Condition C, a 2 x 2 matrix displaying two symbols (and two blank cells), Subject TL averaged 77% correct responding across three sessions. And finally, Subject TL averaged 64% correct responding across three sessions under Condition D, a 2 x 2 matrix displaying three symbols (and one blank cell).

The cognitive demands confronting Subject TL changed systematically access conditions. The task under Condition A required Subject TL to learn that the particular prompting comment from the experimenter was a discriminative stimulus for a specific vocal response. The particular picture symbol on the display also served as a discriminative stimulus for that same vocal response. In function, the experimenter's prompt and the symbol served as "relevant redundant cues" (Engelmann & Carnille, 1982). The "correct" vocal response was operationally defined by the voice template and recognition threshold of the STS/Graphics System, i.e., a correct response was one that was recognized by the system. Feedback to Subject TL on the correctness of her response was in the form of the resultant activation of the speech output or electrical device.

The task under Condition B required Subject TL to transfer her responding to a screen that still displayed only a single symbol at a time, but now the symbols were reduced to one-fourth of their original size and appeared in any of four cell locations on the screen on any given trial. Subject TL made the transition to this new visual interface with little difficulty.

The task under Condition C presented dramatically different circumstances to Subject TL. While the screen continued to be divided into four cells, for the first time more than one of the cells contained a pictographic symbol, i.e., two symbols were displayed at the same time. The symbols would appear in any of the four cells on any given trial. Now a symbol did not function as a relevant redundant cue for correct responding. Simply emitting one of the vocalizations associated with either symbol was not adequate. The only cue available to Subject TL to indicate what her "correct" response should be was the most immediate comment from her communication partner (the experimenter). This approximated the conditions under which a natural conversation might occur. If Subject TL emitted the vocalization that was appropriate to this comment, then correct feedback was provided (in the form of spoken output or device activation). The reason for focusing first on the subject's responding in a conversation rather than initiating a conversation was logistical: the researchers wanted to be certain the subject was not simply emitting, or cycling through, all of the vocalizations that were represented on the display, regardless of their appropriateness or meaning.

Subject TL's performance was not disrupted by these more complex task demands. She had to learn more precise discriminations, i.e., while the two symbols that were displayed served to reduce the range of potential responses from six vocalizations to two, to determine which of the remaining two options she had to attend solely to the comments of the experimenter. Subject TL learned this quite readily.

The task under Condition D presented to the subject more available options for responding. Three symbols were now displayed in various cell locations on any given trial. The subject had to learn that the discriminative stimulus under these conditions was still the communication partner's most immediate comment. During the initial session of this condition (Session 23), Subject TL's performance was disrupted relative to Condition C. She responded correctly only 36% of the time during Session 23. However, she picked up her performance to 94% correct responses on the very next session. During the last three sessions of this research, Subject TL averaged 63% correct responses under Condition D. These data suggest that she not only understood the utility of the STS/Graphics System as a vehicle by which she could communicate, but also generalized this understanding to a wide and unpredictable array of visual interfaces within a 2 x 2 matrix.

Anecdotal data from the experimental sessions provide intriguing suggestive evidence to account for the variability in Subject TL's performance from session to session across conditions. Experimenters repeatedly reported that, as her experience with the STS/Graphics System progressed, she increasingly emitted the vocalizations for her most highly-preferred spoken-outputs/activations, a request for cheese-and-crackers and a comment about nail polish -- even on those trials in which the symbols for these two items were not displayed. In other words, Subject TL's understanding of the purpose of the system and her actual use of the system advanced from that of responding to a partner's questions/directions to that of initiating an interaction. Even though these vocalizations were technically categorized as "incorrect", because they did not respond to the experimenter's questions/directions, the experimenters expressed a high degree of confidence that they were based not on Subject TL's inability to understand the question/direction nor her inability to come up with the "correct" vocalization. Rather, they were a function both of her more complete understanding of the system as the means by which she could express herself in general and her increasing assertiveness in controlling her environment.

These observations suggest a remarkable change in a subject with severe mental retardation. Not only was she generalizing her use of the system far beyond what she had directly been taught, but she also was no longer dependent on the concrete visual referents for her vocalizations. In other words, in typical system use, the symbols function as a type of cognitive prosthetic aid that "remind" the subject of the set of "eligible" vocalizations from which she should choose to emit one. Subject TL progressed past the point of needing this aid and was selecting her vocalization, solely from memory to communicate her intentions. This situation more closely approximated the "give and take" of a natural conversation in which some of the communications are responses to the communication partner, some are initiations. The discriminative stimuli for the particular initiations often include internal stimuli, i.e., internal need states. The researchers are quite pleased to have observed this progression in the subject who was a direct selector. The dynamics of this progression and its relationship to more powerful system-use by subjects with cognitive disabilities will be the focus of future research efforts.

In conclusion, both Subject DL who accessed the STS/Graphics System via scanning and a vocal response and Subject TL who accessed the system via direct selection using vocal responses learned how to use the system appropriately to communicate. They progressed from full-screen single-image displays to 2 x 2 matrices that lay the foundation for greatly expanded and more versatile communication in the future. The reduced cognitive load on system use provided by the graphics visual interface and the reduced physical load provided by the voice interface greatly facilitated their communicative competence with the system. The ease in incorporating personally-meaningful pictographic referents and in configuring screen displays with the STS/Graphics System appear to be major factors in accomplishing these advances for children with severe cognitive and physical disabilities.

References

Alberto, P, A., & Trouttman, A. C. (1990). Applied behavior analysis for teachers. (3rd Edition). Columbus, OH: Merrill Publishing.

Engelmann, S., & Carnine, D. (1982). Theory of instruction: Principles and applications. New York: Irvington.

Smith, D. (1979). The improvement of children's oral reading through the use of teacher modeling. Journal of Learning Disabilities, 12, 172-175.

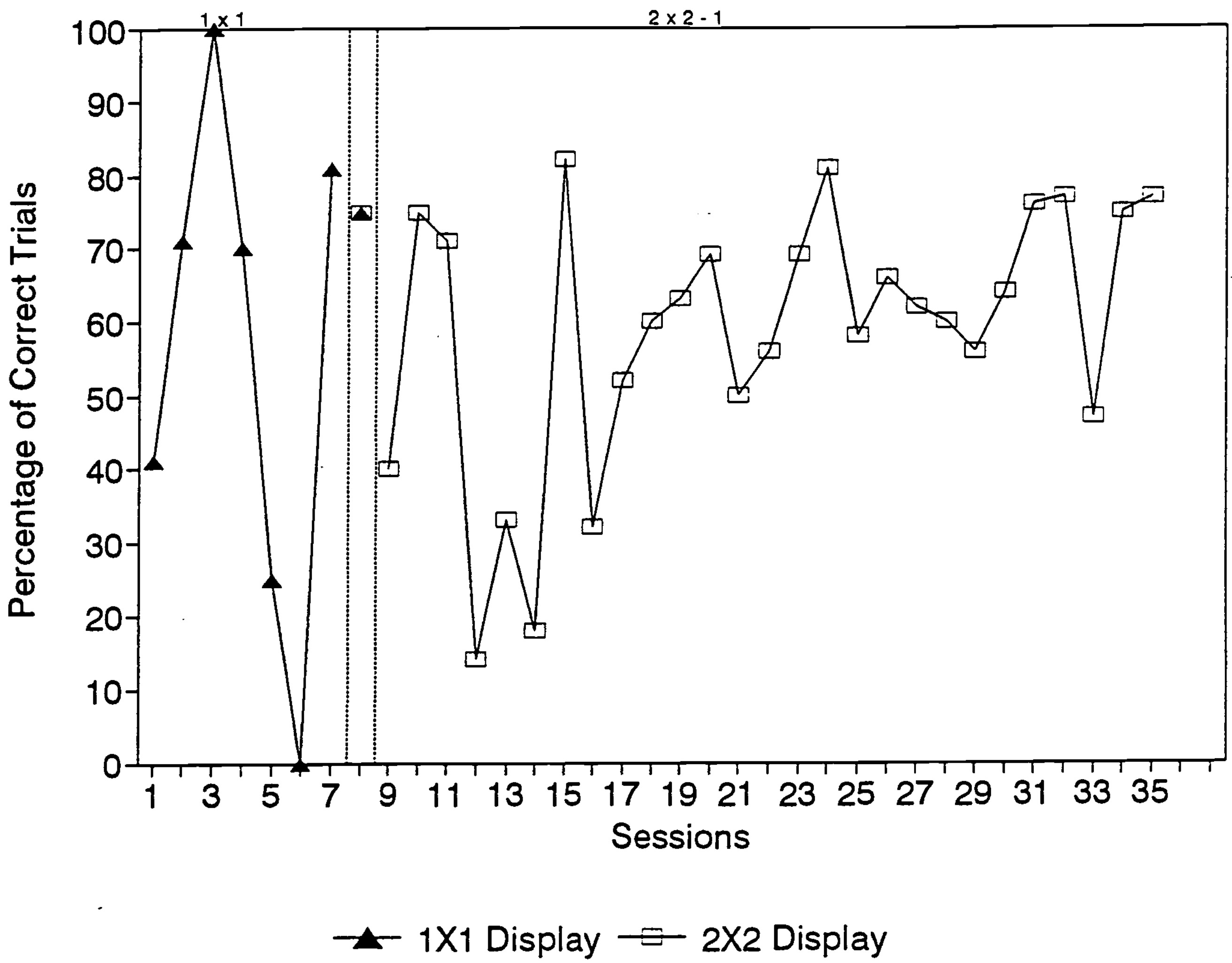


Figure 1. The percentage of trials in which Subject DL responded correctly via the STS/Graphics System as a function of daily sessions.

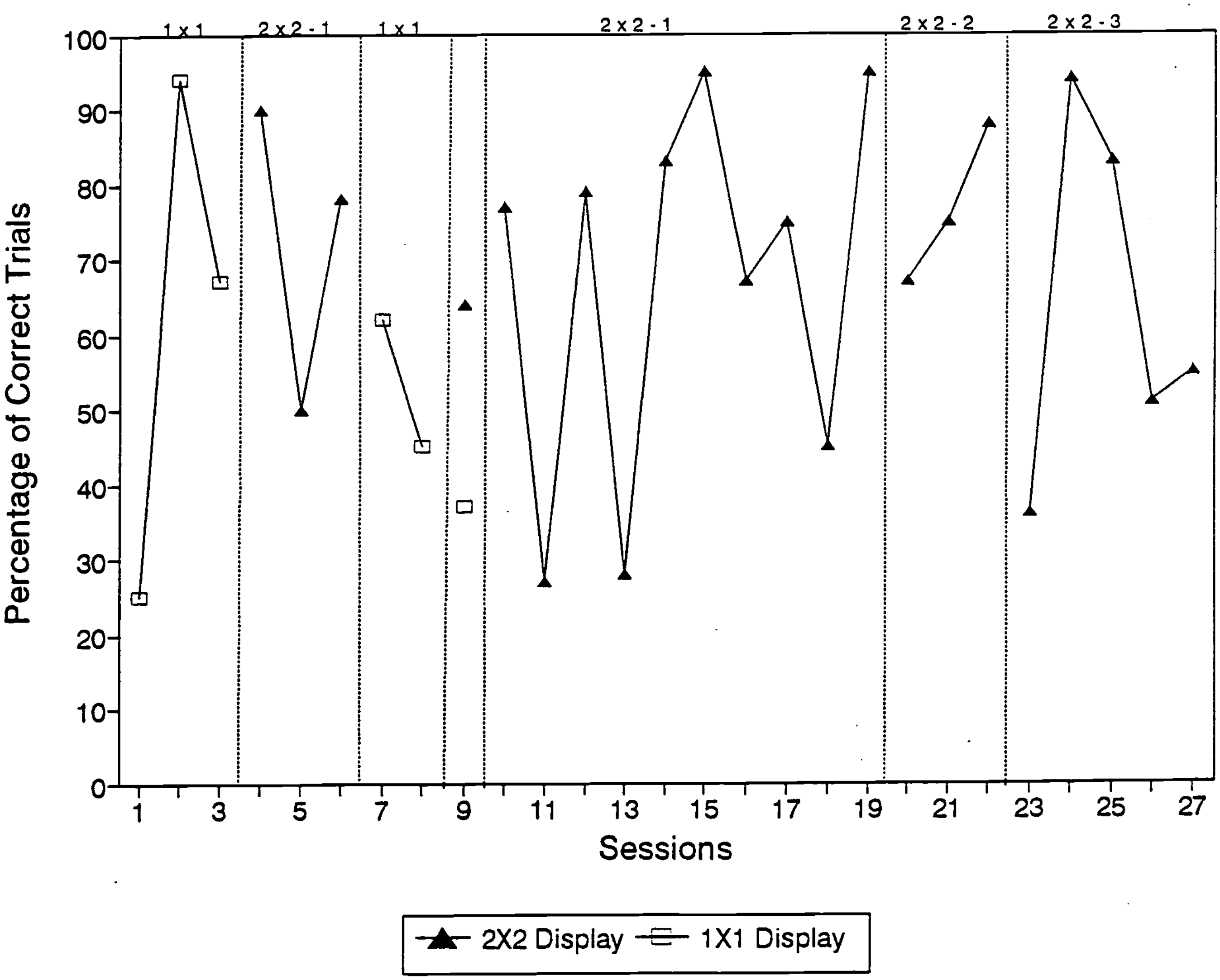


Figure 2. The percentage of trials in which Subject TL responded correctly via the STS/Graphics System as a function of daily sessions.

Sound-to-Speech/Graphics
Initial Subject Profiles

Subject # & Sex	Date of Birth	School	School District	Disability
1 - M	7/11/74	Travis Jr High	Irving	Severe MR, legally Blind
2 - F	9/23/81	A.S.Johnston	Irving	MR (accessed 18-23 months)
3 - M	3/2/77	A.S.Johnston	Irving	MR, Autistic
4 - M	6/9/84	A.S.Johnston	Irving	CP, MR, hyrocephalic
5 - M	3/24/84	A.S.Johnston	Irving	MR, (difficulty assessing - 18 months?)
6 - F	2/24/79	A.S.Johnston	Irving	MR, seizure disorder
7 - F	4/13/70	Irving High	Irving	Unresponsive to accessment (speech 2 months - Binet)
8 - F	5/18/81	A.S.Johnston	Irving	CP, hyrocephalic (shunted)
9 - F	12/28/79	A.S.Johnston	Irving	MR, mild CP (20-30 months - scattered)
10 - F	4/11/74	Irving High	Irving	MR, MH, autistic, neurological problems, (no assessment)
[illegible] - M	9/21/82	A.S.Johnston	Irving	(Assessed 2 years 10 months)
12 - M	4/1/78	A.S.Johnston	Irving	MR, (Assessed 4-6 years)
13 - F	7/19/83	Meadow Creek	HEB	Static encephalapathy with spastic displasia
14 - F	11/7/69	Career Center	HEB	Severe MR, SH, CP
15 - F	1/14/69	Career Center	HEB	OH, MR, CP, SH (2-3 year equivelent)
16 - M	10/18/84	Bell Manor	HEB	OHI, SH, Neuromuscular Desease
17 - M	10/10/78	Career Center	HEB	Autistic, ISH, Leukemia
18 - F	4/5/83	Career Center	HEB	MR, OHI, SH
19 - F	6/8/86	Bell Manor	HEB	MR, SH, CP
20 - M	2/12/83	Career Center	HEB	Down Syndrome, (speech - exp. 4-16mths Rec. 12 mths)
21 - M	9/17/71	Career Center	HEB	MR, OH, SH, CP (no test results)
22 - F	6/10/86	Donna Park	HEB	MR, CP, ISH (getting glasses)
[illegible] - M	5/26/83	Meadow Creek	HEB	OH, SH, CP
24 - M	11/30/69	Career Center	HEB	MR, OH, SH, CP (no assessment)

HARDWARE AND SOFTWARE

The Research and Logistics of Producing the STS/Graphics Package

Hardware Components

Computer: 286 or 386 Central Processing Unit (CPU), 30 MB Hard Disk, Floppy Disk Drive, 640 KB RAM

Super VGA Monitor

Paradise Graphics Board

Microtek Color/Grey Scanner

Votan 2000 Voice Recognition Board, Microphone & Speaker

X-10 Controller

Universal Remote Control

The Computer System

The STS/Graphics system software was developed and operated on either a 286 or 386 CPU. The software has not been tested on a 486 CPU although there are no design constraints to prevent this. Therefore, it is expected that the software would operate within the 486 environment. The software code was developed on three different models of computers: an IBM Computer System with a 286 CPU; a Clone Computer System with a 286 CPU; and a Clone Computer System with a 386 CPU.

Hard Disk Drive for Information Storage and Retrieval

When the computer equipment was purchased for this project, two complete systems were acquired. One computer had a 20 MB hard disk drive and the other had a 30 MB drive. These drives allowed storage of programs, graphics/image files (each file contained the scanned image of one photograph), and data for several students at the same time. Image files varied in size depending upon the size of the image as it appeared on the monitor and the variety of colors in the image. Due to disk space problems, these files were stored on floppy disk rather than on the hard disk.

Image Capturing Device Selection and Its Use to Create Graphic Libraries

For this project, color was an essential part of any image that became a graphic representation in the system. Each image was captured into the computer by digitizing it. Three methods of digitizing images were considered: a scanner, a video camera, or a still camera. Use of any of the three technologies would create a color graphic image of approximately equal quality. The cost, excessive number of components to each system, and complexity of operation for both the video and still camera eliminated them from consideration.

Several scanners were considered, but since color scanning was a requirement, most scanners were beyond the designated hardware budget. After researching all possible devices, a Microtek Color/Grey Scanner, LS-Series, model LS-2400 was selected. *LeColor*, a custom software program, was acquired to interface the scanner and computer since the software which came with the scanner was not able to operate within the requirements for the STS/Graphics project.

Images were digitized and scanned from photographs taken of objects familiar and preferred to project participants. If the image was reduced in size after being scanned, image clarity was reduced and distortion resulted. Consequently, prior to the images being scanned, the display size (full screen, 2X2, 3X3, 4X4 or 5X5) was defined and the image was scanned specifically for that size to help eliminate image degradation. Once the image had been digitized, *PC Paint* software (a drawing package) was used to optimize the image position and to minimize background. Images were stored as separate files on diskettes for future use.

Graphics Monitor for Color Presentation of Stimuli Items

At the outset of the STS/Graphics project, the standard for most high resolution color monitors was a 640 X 480 pixels display (EGA). To provide extremely high quality image displays for the STS/Graphics project, it was necessary to integrate a Super VGA monitor and a graphics adaptor card which had additional memory which would give a video display of 800 X 1000 pixels. Both of these hardware items were new to the market and had little support software. After extensive research, *The PCX Programmer's Toolkit*, of Genus Microprogramming, was identified as providing both the correct video display for 800 X 1080 pixels and the software drivers. This allowed the memory on the video adaptor card to be used to display many high quality images simultaneously on the monitor. The Paradise Graphics chip set was an accepted graphics standard when the STS/Graphics software was developed, so this the graphics software was developed to operate in that mode.

The Votan 2000 Voice Recognition Input/ouput Board Serving the Voice Recognition and Speech Output Function

The Votan 2000 Voice Recognition Input/output board, which is housed internally in the computer, had been used by the Bioengineering Program in a previous speech recognition project. At the beginning of this project, new voice recognition technology was researched and it was confirmed that the

Votan 2000 was still one of the most reliable voice recognition boards available. Based on this finding, the existing boards were used in this project.

Votan was in the process of developing software libraries for operation with their series of voice boards. The Bioengineering Program waited for the completion of these libraries to integrate them with the STS/Graphics software. When the libraries were delivered, it was found that they had been developed for only one C compiler and that they could not operate in the development environment being used by the programming staff. Several weeks were spent working with Votan to evaluate whether they could modify their source code to operate with the various compilers. It was concluded that it would be several months before Votan could possibly make the necessary changes and that it was not a priority activity for them. Once this was known, the Bioengineering programming team rewrote software code from the Pascal language which was the language used for the previous Bioengineering Program voice recognition project. During this process, the software code was converted to "C". The software code eventually operated both the voice recognition and voice record functions needed for this project.

Microphones and Speakers for the Voice Input/Output to the Votan Board

The Votan 2000 Board is supplied with a microphone and speaker unit. The speaker measures approximately 2 X 3 X 4 inches, and the microphone is mounted on a gooseneck on the top of the speaker. Both components are attached by audio cables to the Votan board which is housed in the computer. During the evaluation of the subjects, it was found that in some cases the microphone/speaker was a distraction to the subject. Unsuccessful attempts were made to position this unit in a less obtrusive location. While searching for an alternative to this situation, a variety of microphones were tested for use with the Votan 2000 Board. These included a conference microphone manufactured by Realistic, which appeared to be a flat sheet of metal approximately 4 X 5 inches. Even though the subjects were unaware of this unit, this microphone was unreliable when interfaced with the Votan Board. A second Realistic microphone, which was designed for use in presentations, was tested. This unit was very small and clipped onto the users clothing on a collar or somewhere that was close to the users' mouth. Performance for this microphone with the Votan 2000 was acceptable and students were less aware of it than they had been with the Votan microphone. In some cases, subjects preferred a researcher to hold the microphone which was not a distraction. Additional microphones were tested, but none were able to maintain high enough quality of speech to interface satisfactorily with the Votan 2000 Board.

When a microphone other than the Votan was used, a Realistic speaker was interfaced for speech output. This unit provided greater volume than the Votan unit and in time it replaced the Votan speaker even when the Votan microphone was used.

X-10 and Universal Remote Control for Environmental Control Function

X-10 controllers, which are electrical modules for environmental control, are supplied with a serial interface and software drivers for computer operation. Bioengineering programmers accessed the X-10 source code and converted it to the "C" programming language. This code operated reliably on an IBM PC/XT computer, but when it was transferred to a faster CPU, reliability problems were experienced. The development team worked extensively with X-10 technicians to resolve this problem. The first identified problem was that the speed at which the signal was sent from the computer to the X-10 controller was too fast. This was resolved by slowing down the signal.

Further problems were experienced during the software development phase when code to operate the X-10 Controller Unit was integrated with other modules of the STS/GRAPHICS software. The X-10 operated unreliably. Problems ranged from X-10 failure to complete serial port failure. The difficulty was overcome by using *Greenleaf Software Library* routines for port control. Because the problem seemed corrected, it was assumed that the problem had been solved. It was found later that the "solution" had only solved the obvious problem and not the underlying cause.

The underlying cause for these intermittent failures were traced to hardware difficulties. After having three serial ports replaced on various computers, it was found that there was an incompatibility between the Votan 2000 voice input/ouput board, utilization of the computers serial port, and the computer chip set. If the Votan board was operated while the X-10 signal was passed through the serial port, then X-10 failure was experienced. The types of failures depended upon the X-10 instruction that was sent, but all failures were irreversible. The problem was solved after extended trouble shooting by only operating the software on a computer system that had a chip set other than "Chip" brand chips. The ultimate result of this finding was that computers built with this chip set had to be avoided for operation of this software.

The X-10 controller can only operate appliances through house wiring. Thus it was necessary to integrate a method to control devices which operate through infrared control where house wiring is not used. The One-for-All Universal Remote Control unit was the only infrared controller available with an RS232 interface at the time of this project, so was selected to control all infrared operated devices. The One-for-All was sold with software to interface with the RS232 port. Unfortunately, after the product was shipped, it was found that the software was not functional. Universal Electronics, who developed the One-for-All, lost the source code for their software and were unable to assist us in making the RS232 port functional. After researching the problems, operation of the unit was accomplished using original source code developed by Bioengineering programmers which was integrated with *Greenleaf* libraries.

<u>Single Switch Control for Alternate Computer Access</u>

Single Switch Control was accomplished using a PC-Pedal switch which was manufactured and sold by Brown and Company, Inc. This was the least expensive single switch that operated on IBM computer using the RS232 port.

STS/Graphics Development Software

Software development was done using Borland's Software Development Package which includes the Turbo C compiler, the programmers tool kit, and the debugger. It was found that in most instances, this development environment was compatible with other resource libraries.

STS/Graphics Custom Software Development

The STS/Graphics system operating software was designed to be modular. By developing the software in a modular format, it allowed for the independent development of the various sections of code. Each section of code was written, tested, debugged and then integrated into the total software package. Major modules were for voice input/output, scanning, graphics displays, environmental control, data collection and retrieval, report generation, and STS/Graphics operation.

The scanning and graphics modules were developed simultaneously followed by the voice input/output module, environmental control, and STS/Graphics operation. The final module which was developed was data collection and reports. These modules were dependent upon all other modules for completion.

Impact on the Research Project

Delays in completion of the software were due to ongoing hardware problems and support software failures (see hardware section). Consequently, initial evaluation and intervention with the subjects were initiated with the scanning and graphics module and a intermediate version of the voice input/output module. The voice training portion of the software which display full screen images of objects was added to the project in August, 1990.

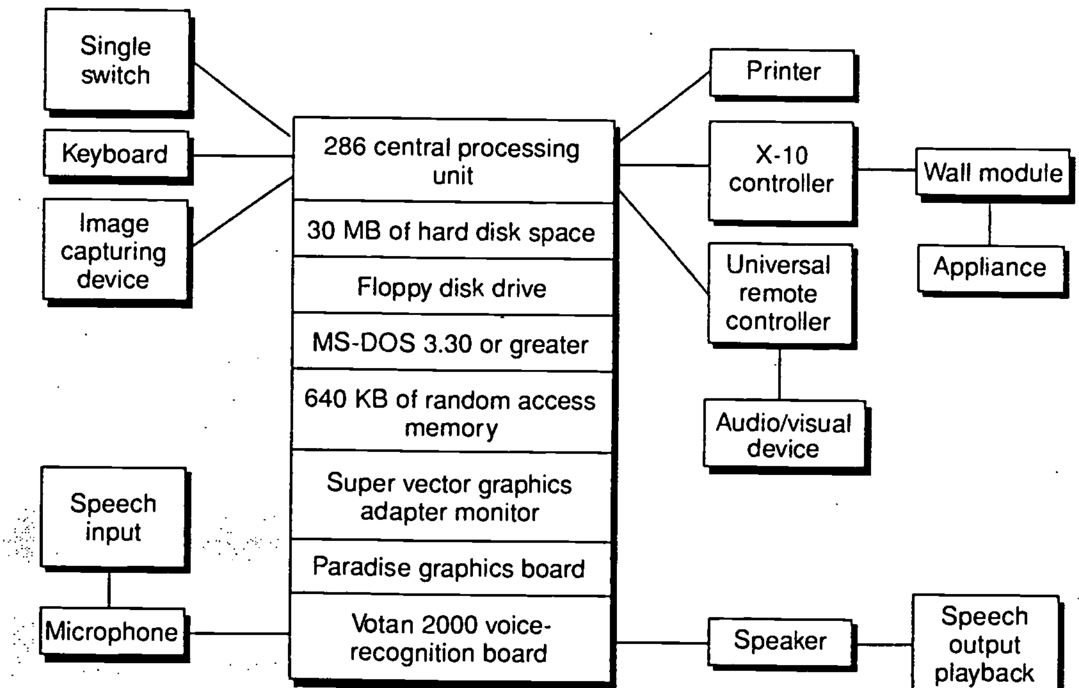


Figure 1. Sound-to-Speech/Graphics Hardware Configuration

Description of the Software and Hardware Development and Design

The research described herein involves the design and development of a voice-operated AAC computer system that was used by children and young adults with mental retardation to communicate and to control their environment through no greater effort than the vocalization of simple sounds.

The STS/Graphics System is designed to research the potential of voice recognition technology for persons with mental retardation and is an extension of previous research and development conducted by Brown and Cavalier and their colleagues. In earlier research, a voice-activated computer system was developed and evaluated with a woman diagnosed with mental retardation and severe multiple disabilities, who needed pervasive support. The woman learned to voice activate the system to control electrical devices in her immediate environment.[1,2] This was followed by research on a multi-user system with environmental control and speech output capabilities.[3] Neither of these studies employed a picture-based display to assist the user in making choices.

The next generation in this research track, the STS/Graphics System, incorporates a graphics display. Key features of the STS/Graphics System are as follows: (a) photographic-quality computer-generated images of items/persons in the user's natural environment that are presented on a display to represent choices among spoken communications and environmental control activations; (b) vocalizations, whether intelligible or not, that are used to select choices from the image display; (c) physical switch activations and traditional keyboard inputs that are alternate access modes; (d) voice and keyboard access modes that can be used to make choices via direct selection; (e) voice and switch input that can be used with linear scanning and row/column scanning; (f) speech output that is in the form of digitized, easily-understood speech and that is age and sex appropriate; and (g) data on system use that is automatically recorded, thereby facilitating the development of appropriate training strategies. *Figure 1. Sound-to-Speech/Graphics Hardware Configuration* depicts the hardware components of the STS/Graphics System.

A more detailed description of the STS/Graphics System follows:

Hardware

The following list of hardware components represents the minimum configuration required for the operation of the STS/Graphics System. Enhanced performance can be obtained with a 386 or 486 microprocessor and larger hard disk capacity. Additional Random Access Memory (RAM) will not enhance performance.

Hardware Components

- o International Business Machine (IBM) Personal Computer (PC) Advanced Technology (AT) or compatible with a 286 central processing unit (cpu)
- o 30 Megabyte (MB) of hard disk space
- o Super Vector Graphics Adaptor (VGA) monitor
- o Paradise Professional Video Adaptor
- o Votan 2000 speech recognition board
- o Image capturing device with color capability
- o X-10 Powerhouse and various wall modules
- o 640 Kilobyte (KB) of RAM
- o Floppy drive
- o Microsoft-Disk Operating System, version 3.30 (MS-DOS) or greater

Voice Input

The voice recognition system incorporates the Votan 2000 voice recognition circuit board manufactured by Votan Inc. of Fremont, CA and software routines unique to the system developed at The Arc. When operating the STS/Graphics system, the user's voice functions as a switch and the user "trains" the computer by speaking the target vocalization into the microphone. This sound is recorded digitally on the hard disk. The training is repeated three times and each vocalization is recorded separately. Once the three trainings are stored in a user-specific voice template, the STS/Graphics software analyzes the three recordings and averages them. For scanning access to the system (for more information on scanning, see discussion in Access Mode section), the user specific template is comprised of trainings for one unique utterance. For direct selection access to the system (for more information on direct selection, see discussion in Access Mode section), the user specific template is comprised of multiple unique vocalization trainings. Once all of the trainings are stored, the system checks to verify that the user's target vocalization will successfully match one of the sounds stored in the voice template. If the match is not similar enough to the original trainings, then the user must retrain the voice template for that particular utterance.

When the user is actually operating the system and produces a vocalization, the system checks to see if the vocalization achieves a "match" with the trained vocalizations stored in the voice template. In the event that there is a match, the system will acknowledge the vocalization by activating the appropriate speech output message or environmental control function; in the event of the vocalization not matching any of the trainings closely enough, the system will not respond. If the system does not respond, two optional corrective actions can be taken. First, the accuracy used by the system to match a vocalization to the stored voice trainings can be made less sensitive so that the system is less demanding in its matching requirements. Second, the user may choose to repeat the training process to refine a vocalization stored in the template for better recognition accuracy.

Sound-to-Speech Translation and Expansion

For communication purposes, each user can access speech output messages stored in digitized files on the hard disk. Each message can consist of speech of up to 8 seconds in duration, thus permitting single words or complete sentences to be output. Someone whose voice is age and sex appropriate for the user records these messages into the system. The Votan board manages this process.

The system uses an algorithm to compare a user's vocalization with those previously stored in a voice template. If a match is found, the input is identified and linked to one of the speech output messages. In a sense, the user's often brief and unintelligible vocalization translates into intelligible speech and expands into a phrase or complete sentence. For example, if a user's vocalization for "water" approximates "wuh", the system can immediately output "Could I have a drink of water, please." Message files for many different users can be stored and accessed via one computer system. The number of available files is dependent only upon the size of the system's hard disk.

Environmental Control

The system incorporates X-10 Powerhouse environmental control technology manufactured by X-10 Inc. of Northvale, NJ. The computer sends digital encoded signals over existing electrical wiring in the room or building in search of a lamp or wall "module" with the same identification code. The modules plug into wall outlets. The target module responds to the control signals transmitted from the computer and activates or deactivates the appliance plugged into it. Any voice (or other) input can be linked to the activation and deactivation of designated electrical devices. A single voice input can also be linked to any combination of speech outputs and device activations, e.g., the vocalization "pa" translates into "I'm going to watch TV now", and the television set turns on.

The system also uses a One-for-All universal controller manufactured by Universal Electronics, Inc. of Signal Hill, CA, to control any audio/visual device that operates through the use of an infrared remote controller. The universal remote controller operates via a serial I/O port located inside its battery compartment. Using a simple communications protocol, the computer system, through an RS-232 port, electronically "presses the buttons" of the remote control unit to operate the device.

Graphics Interface

The system displays on the video monitor photographic-quality images of objects, appliances, and people in the user's environment. The size of the images, as well as the number of images appearing on the display, are customized to the user.

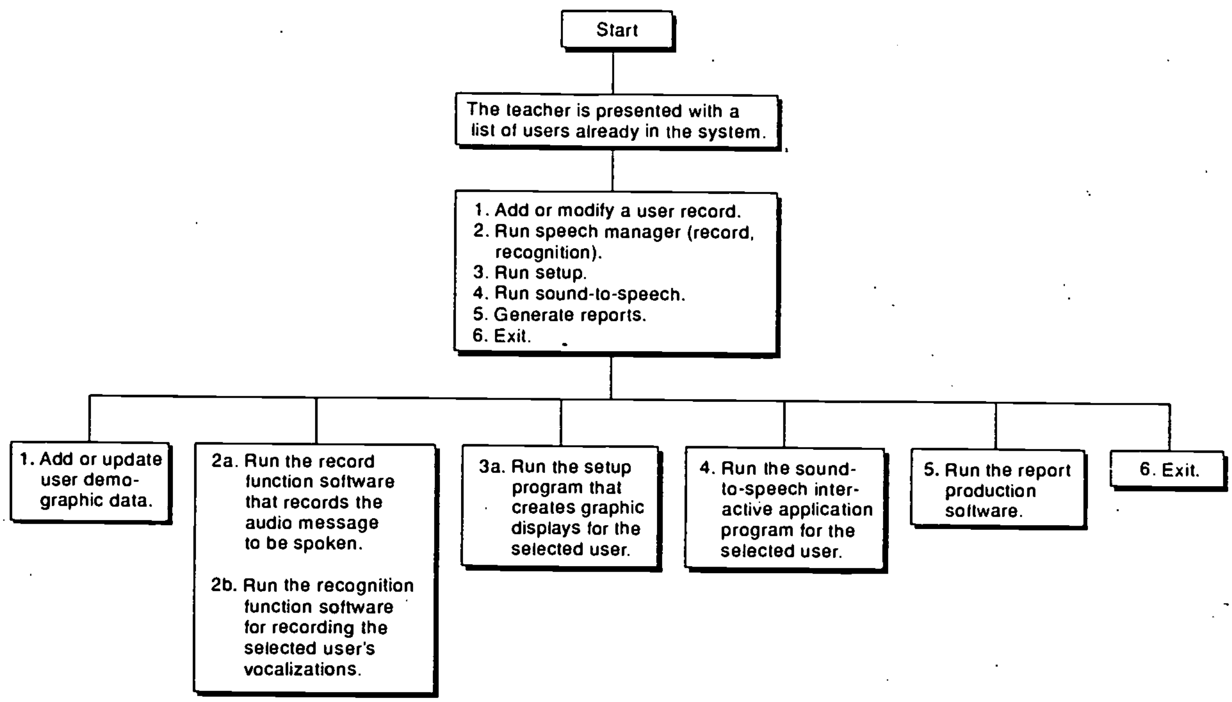


Figure 2. Overview of the Main Menu for Sound-to-Speech/Graphics Software

This is achieved by digitally scanning photographs of items into the computer using an image capturing device (a scanner or video camera) and then sizing the image using image manipulation software such as PC Paintbrush. Extensive image libraries are created for the users. An image library consists of disks containing scanned images of items that have personal meaning to the user. The library is arranged by category and serves as an organizational model for the teacher or trainer. In this way, the same images are used by a variety of subjects in their customized displays.

Displays range from a single image to a 5 x 5 matrix of images. On a 13 inch monitor, the size of each of the four images in a 2 x 2 matrix is approximately 3.75 X 5 inches; the size of each of the 25 images in a 5 x 5 matrix is approximately 1.5 X 2 inches. The graphics display can also "flip" to a second page of choices for each user. The two pages of images permit the configuration of a hierarchical system of choices for users who understand the concept of categorical clustering. In such a system, the images displayed on page one represent superordinate categories, e.g., an image for the "food" category, an image for the "clothing" category, an image for the "sports" category, etc. The selection of a page-one image displays a second page of images all related to the particular category selected on page one. However, it is not mandatory that images on page-one represent super-ordinate categories for the second page. On the second page, one cell is always designated as a "return" to page one.

The graphics images in this system act as cognitive prosthetic aids that serve to remind the user of the speech outputs and device activations that are available. While a user's voice input (or switch or keyboard input) initiates the sequence of electronic events that results in some functional output, the graphics images provide a symbolic representation of output choices that are "live", or available, at any one time.

Access Modes

The STS/Graphics software provides two methods of accessing the system: direct selection and scanning. In the direct selection mode, a user vocalizes a different sound for each image that is presented on the display. The scanning selection mode is operated in two ways: linear scanning or row/column scanning. In either scan mode, the system requires only a single vocalization to access all choices and the vocalization does not have to be consistently pronounced from one occurrence to the next. The vocalization serves to stop the scanning sequence. Once the scan halts, the functions execute that are associated with the image on which the scan stopped (that is, speech output, environmental control, or both). Alternately, the scanning can be stopped by using a single switch.

Software Design

The customized system software is made up of four major components: Speech Manager, Setup, Sound-to-Speech, and Reports as shown in *Figure 2. Overview of the Main Menu for Sound-to-Speech/Graphics Software*.

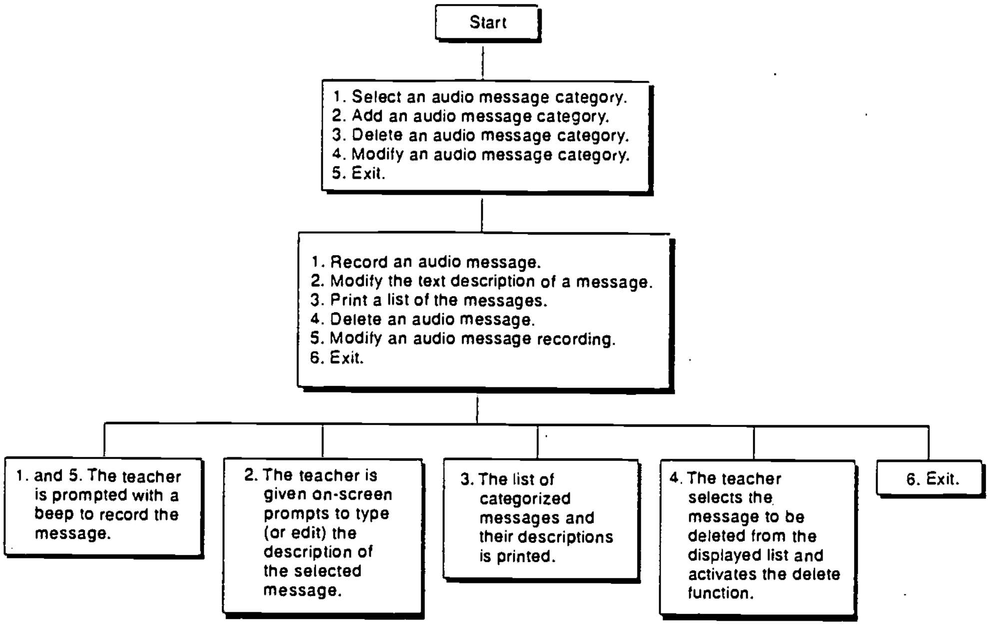


Figure 3. Overview of the Speech Manager: Recognition Function for the Sound-to-Speech/Graphics Software

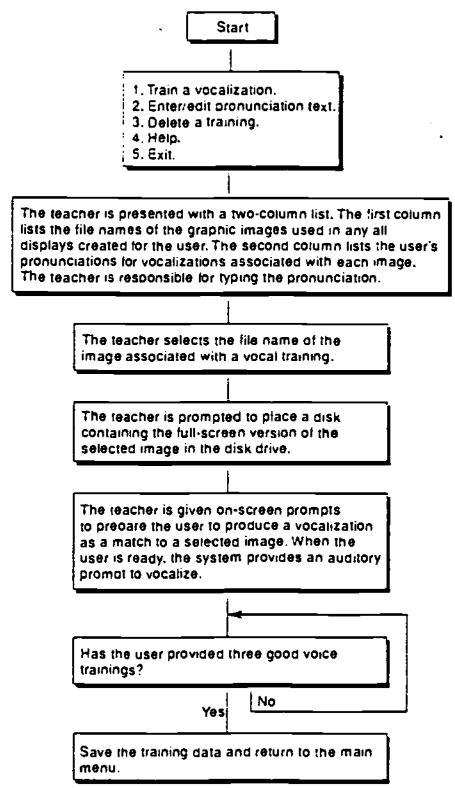


Figure 4. Overview of the Speech Manager: Record Function for the Sound-to-Speech/Graphics Software

48

Speech Manager

The **Speech Manager** has two discrete functions: the Record Function and the Recognition Function:

Recognition Function, as described in *Figure 3. Overview of the Speech Manager: Recognition Function for the Sound-to-Speech/Graphics Software*, creates and manages the voice template that contains the user's recordings of single or multiple vocalizations. In the direct selection mode, a different vocalization is linked to each image on the display. In the scanning access mode, only one vocalization is stored in the system and serves to halt the scan sequence on the desired image, resulting in a selection.

The Record Function, as described in *Figure 4. Overview of the Speech Manager: Record Function for the Sound-to-Speech/Graphics Software*, creates the speech output message files and groups these individual audio messages into categories appropriate for the user or setting. New categories and messages can quickly and easily be added or deleted. Each audio message has a corresponding text description in its storage file.

Setup

The **Setup** component of the software, as described in *Figure 5. Overview of the Setup Function for the Sound-to-Speech Graphics Software* (illustrated on the following page), is used to combine image files, audio output messages, environmental control, and infrared commands and to assign them to specific cells in a matrix (from 1 X 1 to 5 X 5). When all of the desired cells in a matrix are filled with images (it is not mandatory to put an image in all cells), then the information is stored as a "page" file. Each page contains different graphics choices and the user can "flip" between two pages as explained in the Graphics Interface section.

Sound-to-Speech

Once the system has been configured for use, the **Sound-to-Speech** component presents the pages to the user as described in *Figure 6. Overview for the Sound-to-Speech Function of the Sound-to-Speech/Graphics Software* (illustrated on the following page). It is the interactive portion of the system and the only one with which a user interacts for communication and environmental control. Data collection on user activity occurs whenever Sound-to-Speech is active. The collected data can be output as a report.

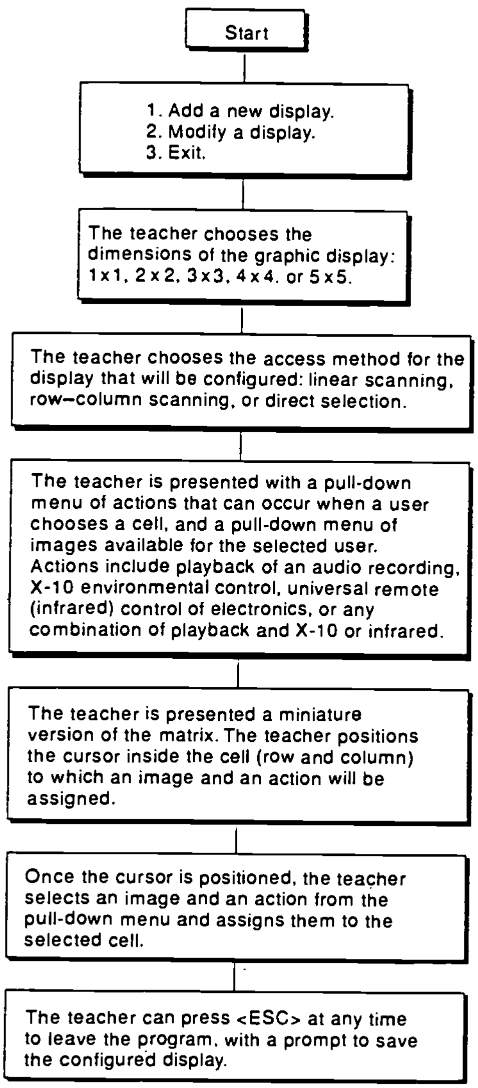


Figure 5. Overview of the Setup Function for the Sound-to-Speech Graphics Software

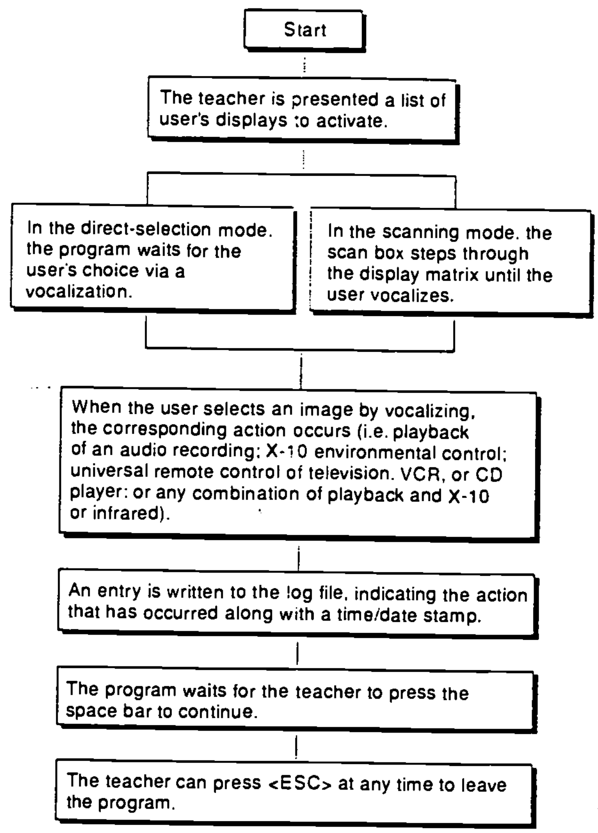


Figure 6. Overview for the Sound-to-Speech Function of the Sound-to-Speech/Graphics Software

BEST COPY AVAILABLE

Reports

The system generates activity reports regarding the operation of the system. As each user's choice is activated, a record of the interaction is written to a data file. This consists of the time of day, the number of the cell selected on the matrix display, the name of the image displayed in that cell a that time, and the text of the speech output or the name of the environmental device that is activated. The report also includes demographic information on the user, the access mode, the name of the particular matrix display, the dimensions of the matrix, the sensitivity setting on the voice recognition board, the matrix and scanning box (if one is used), colors, the names of the various images and their cell locations, their associated speech outputs or device activations, and phonetically-spelled approximations of the user's voice inputs for the images. When the teacher or user so desires, the software can aggregate and analyze these data on the user's activities and can generate on-screen and/or hard-copy print-outs.

References

[1]Brown, C. C., and Cavalier, A. R., "Voice Recognition Technology and Persons with Severe Mental Retardation and Severe Physical Impairment: Learning, Response Differentiation, and Affect," *Journal of Special Education Technology* (in press).

[2]Brown, C. C., Cavalier, A. R., and Tipton. L., "Increased Independence Through Computerized Voice Recognition for Persons who are Severely Physically Involved and Profoundly Mentally Retarded," in *Proc. Ninth Annual Conference of the Rehabilitation Engineering Society of North America*, RESNA, Washington, DC, pp. 101-103 (1986).

[3]Brown, C. C., "Research Focusing on Freedom of Choice, Communication, and Independence Using Eyegaze and Speech Recognition Assistive Technology," in *Proc. of the First South Central Technology Access Conference*, University of Arkansas at Little Rock, Little Rock, AR, pp. 27-34 (1989).

THE

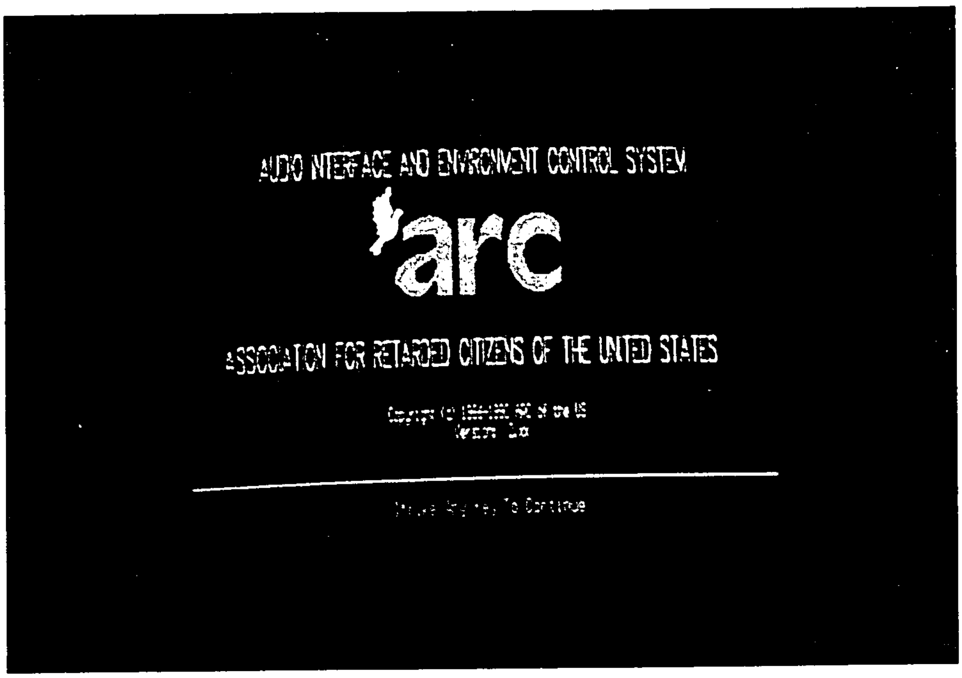
SOUND-TO-SPEECH

SOFTWARE

The Bioengineering Program
Association for Retarded Citizens of the United States
500 East Border, Suite 300, Arlington, TX 76010
(817) 261-6003

August 1991

The Sound-to-Speech Software



a communication and environmental control system using a graphics interface

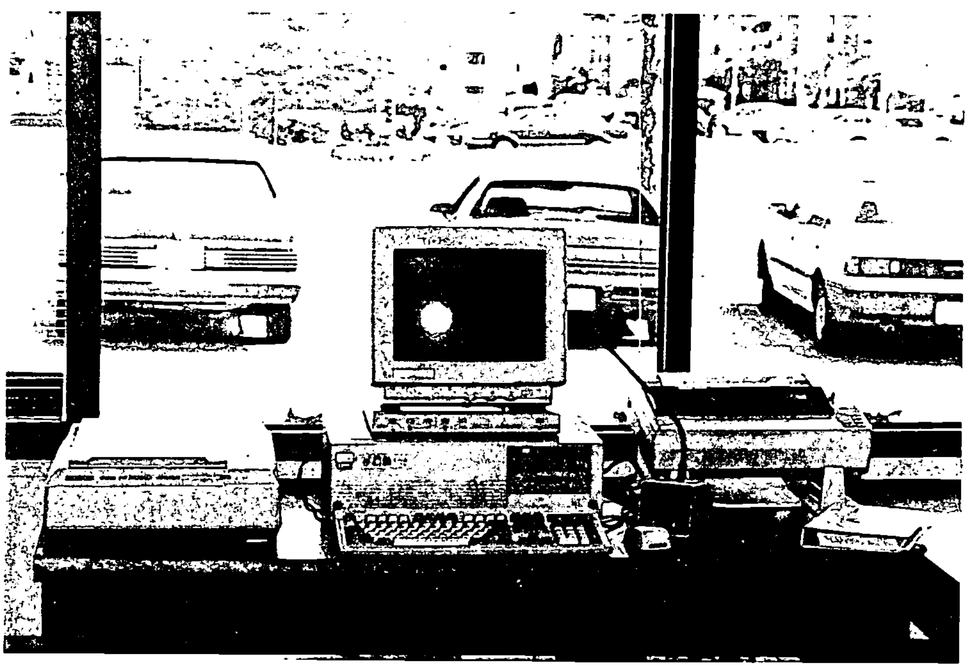
HARDWARE REQUIREMENTS:

PC/AT Microcomputer with 640 Kbytes of memory and a 30 Mbyte hard disk drive

A VCA color monitor

A color scanner

A mouse



SYSTEM FEATURES:

Produces digitized speech

Activates/deactivates small appliances and toys

Uses selections made from graphics screen display

Selections activated by users vocalization/s

Vocalization does not have to be recognizable speech

ADDITIONAL SYSTEM FEATURES:

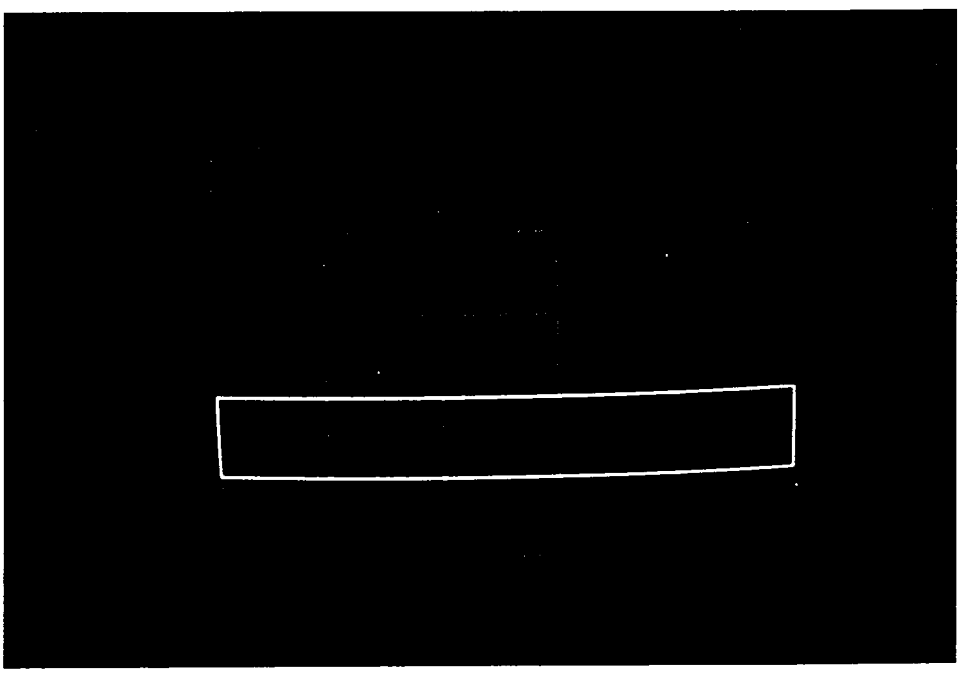
Produces photographic quality graphic images

Images can be rapidly scanned into the system

Customized graphics screens for users

Multiple access modes:

Scanning (row/column. continuous linear, direct)
Direct Selection (requires range of vocalizations)



GRAPHICS CONFIGURATION

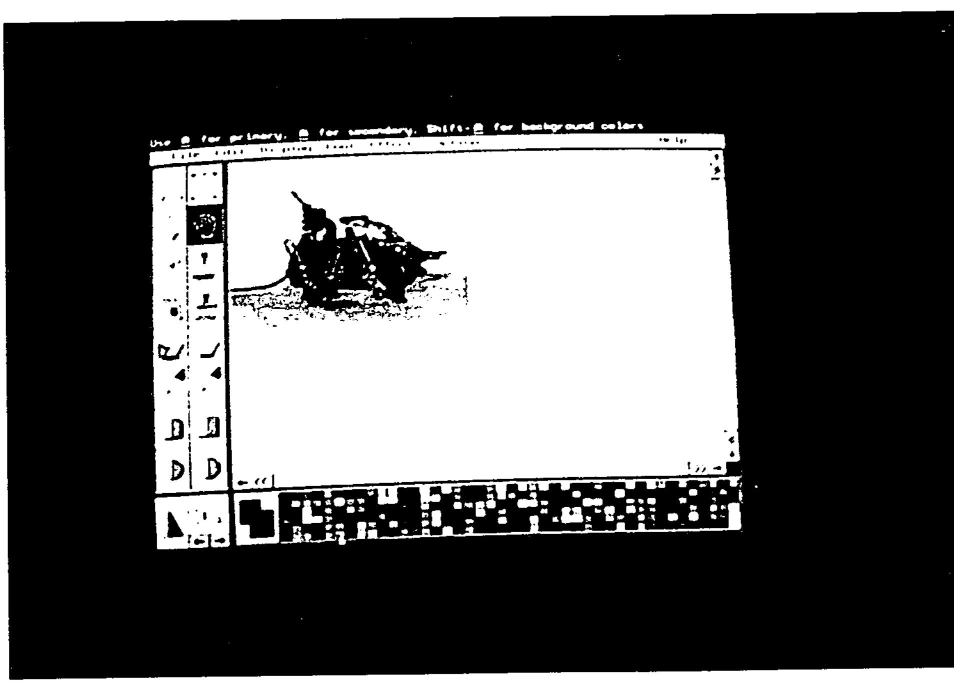
Variable matrix size

Multiple access modes

Picture location in matrix customized to user

Activity associated with each picture

(e.g. picture of light; turns on the light and produces speech output "I want the light on")

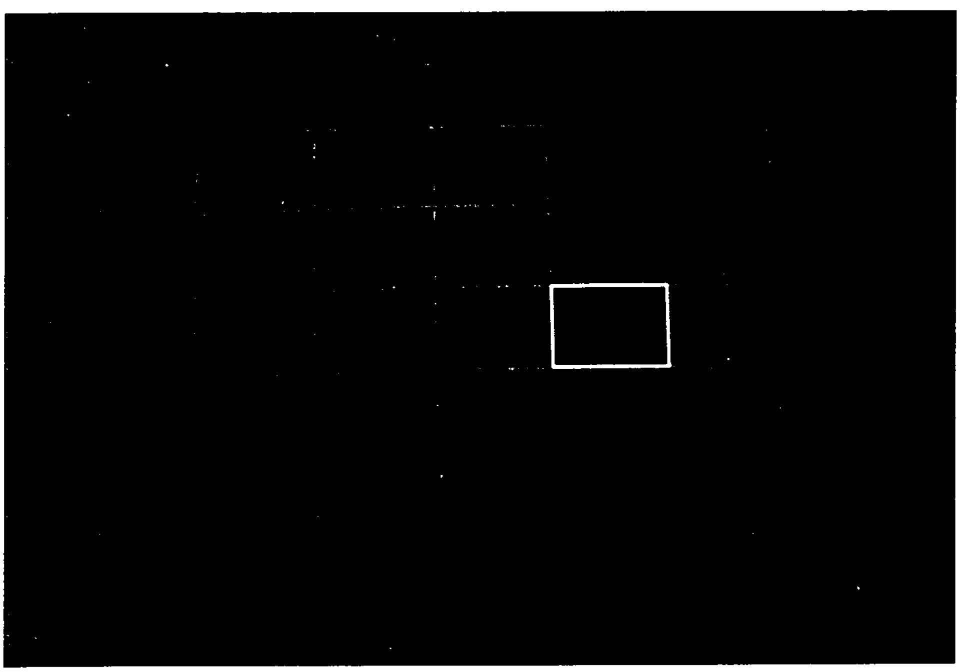


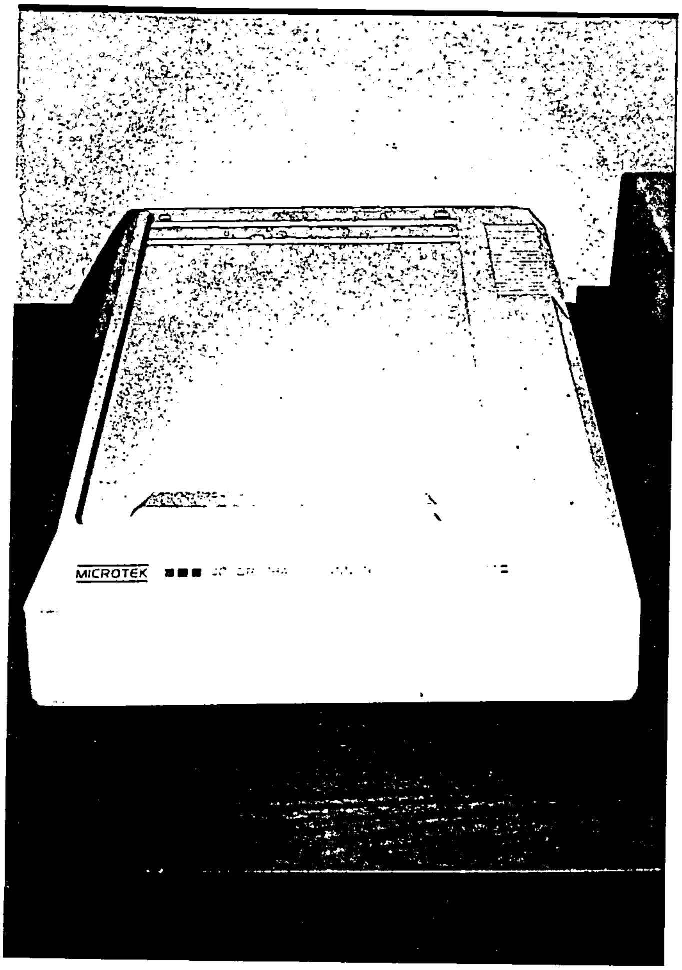
GRAPHICS DISPLAY

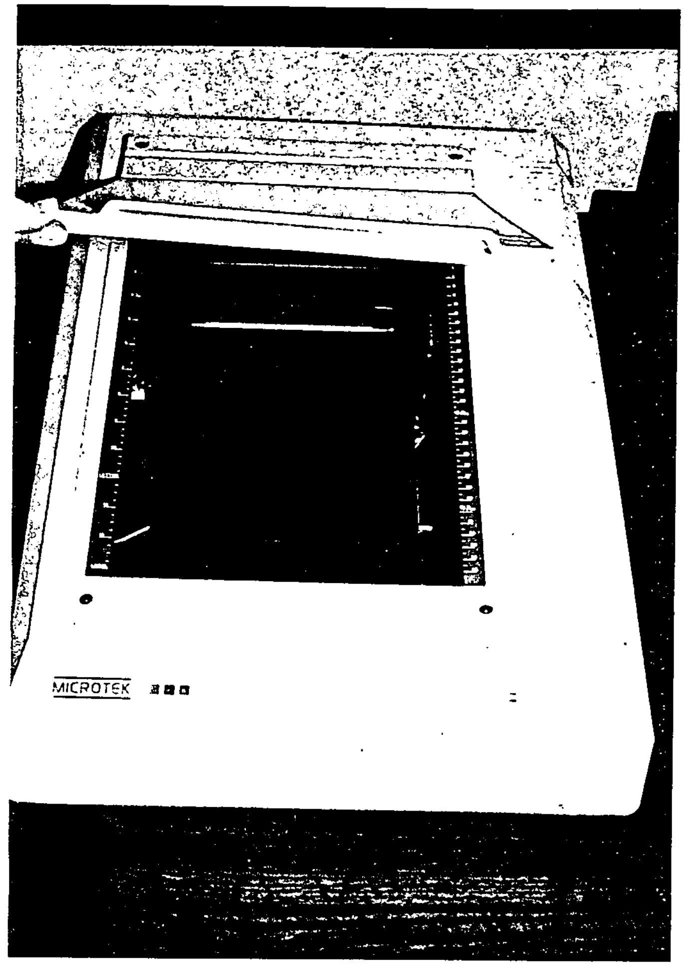
Allows a matrix of 4 to 25 pictures

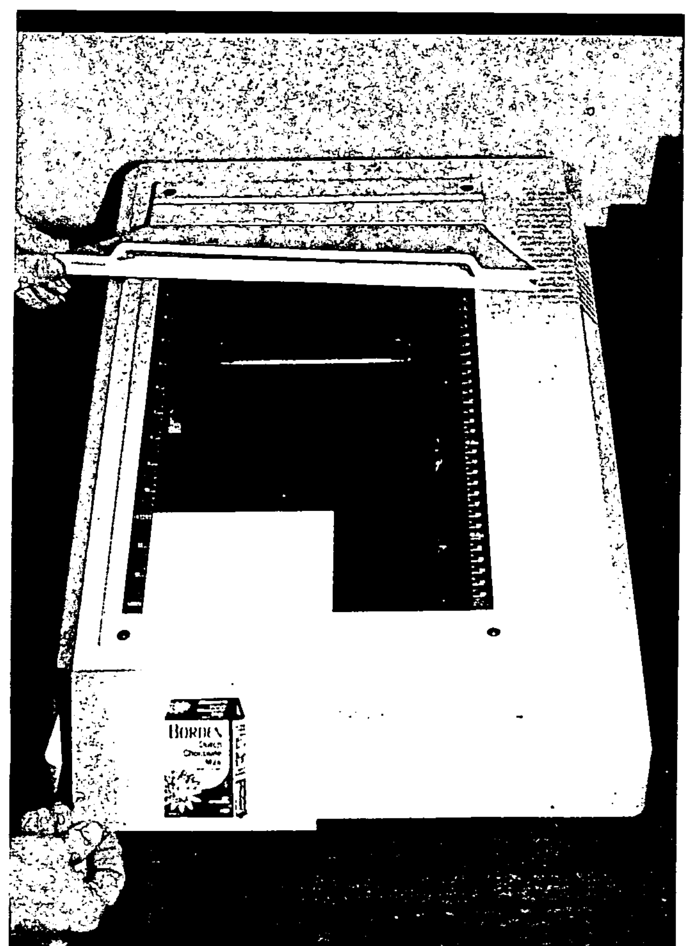
(2 x 2, 3 x 3, 4 x 4, or 5 x 5)

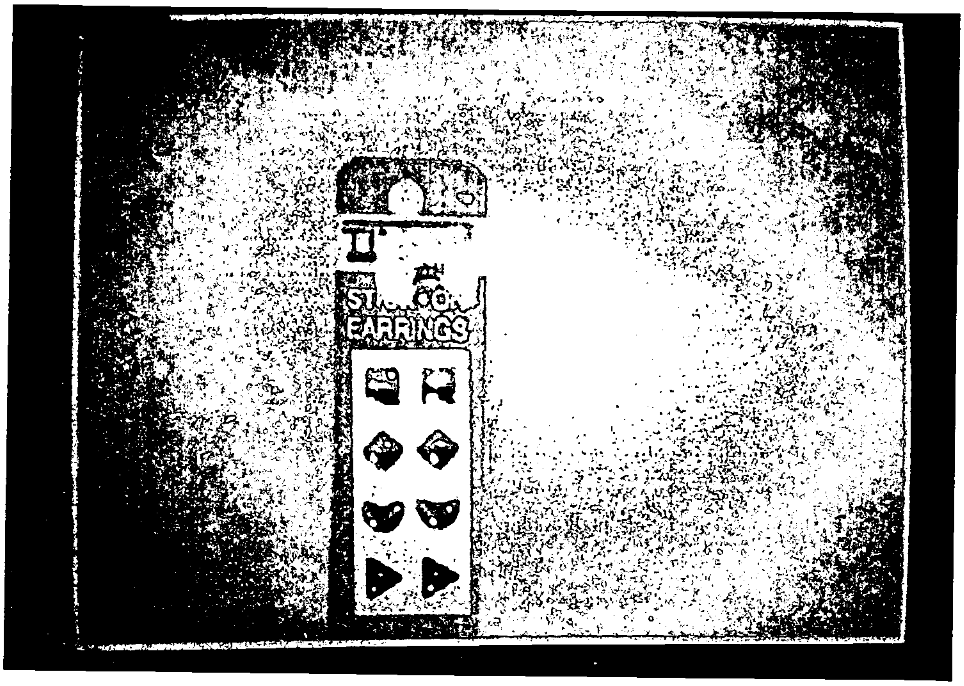
Up to 4 levels of screens available

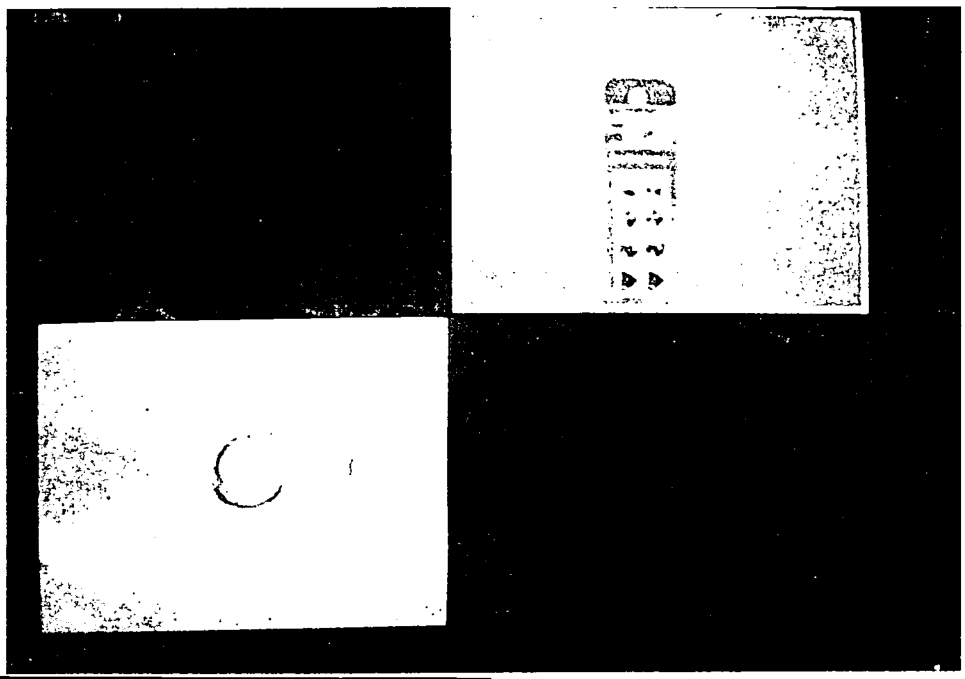


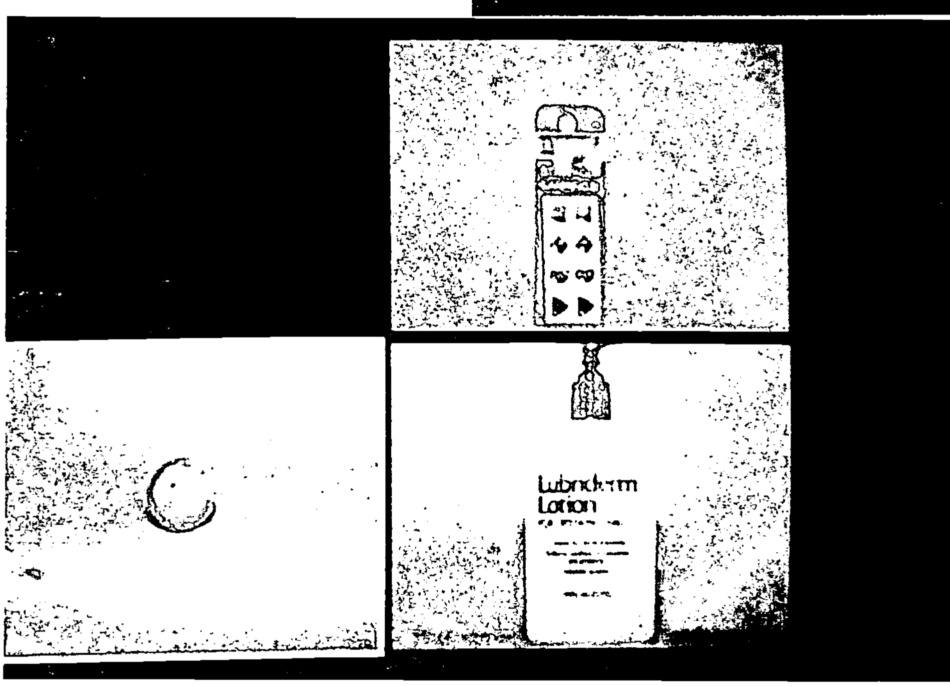

MICROTEK

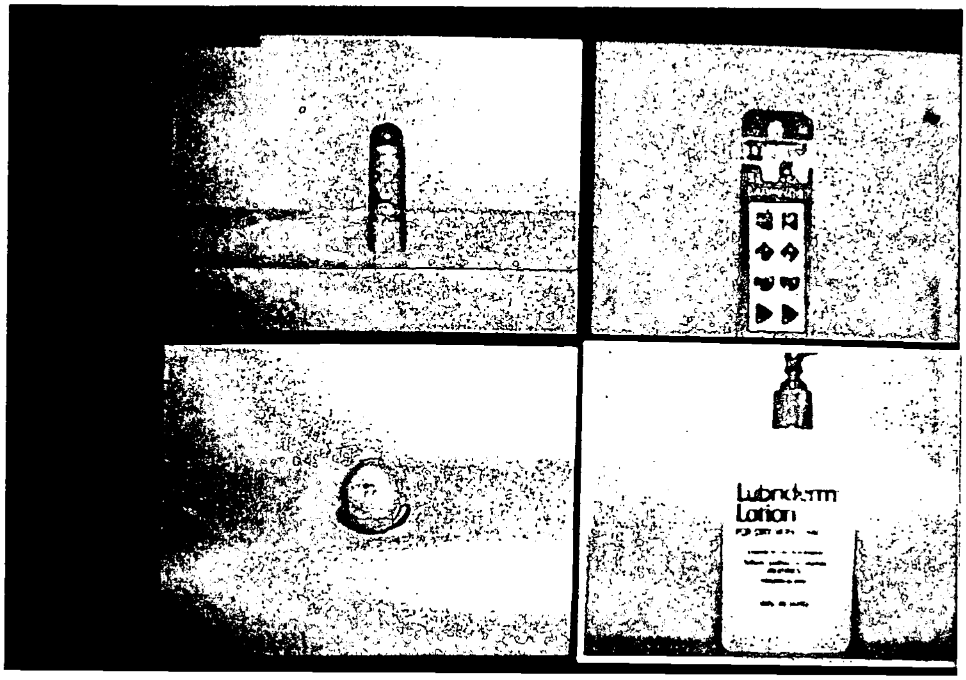

MICROTEK

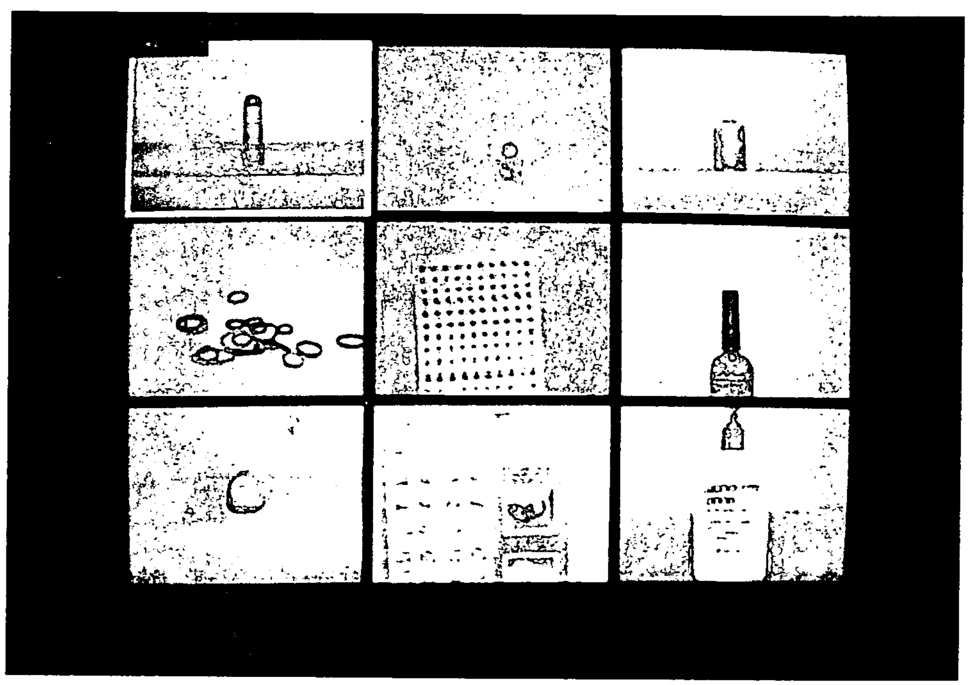

BORDEN


EARRINGS

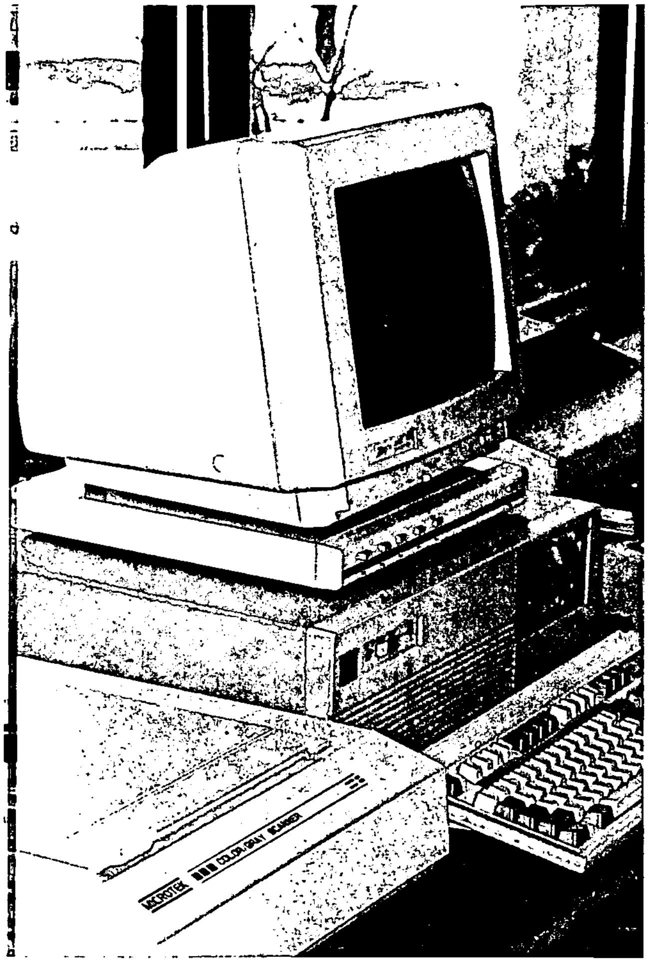


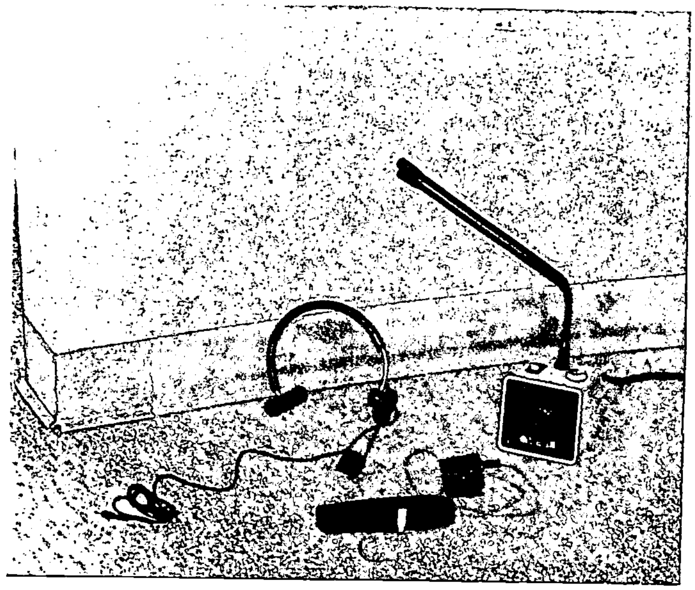

Lotion

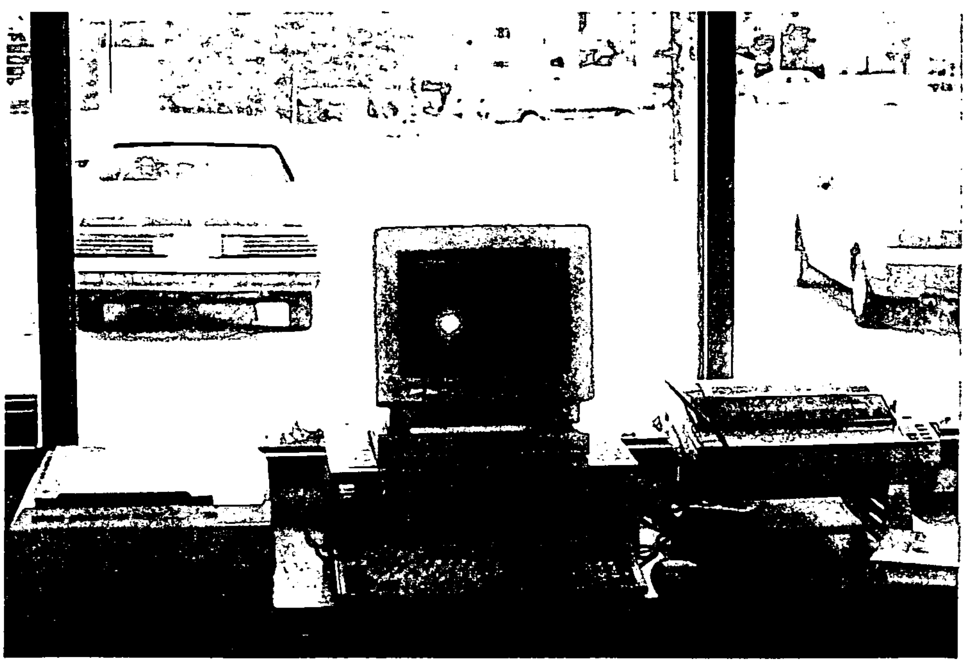




BEST COPY AVAILABLE


MICROTEK
COLOR/GRAY SCANNER





KEY CHARACTERISTICS OF USERS

Unintelligible speech

Limited cognition

Severe physical limitation

Unable to use standard mechanical adaptive switches

Mailed 12/6/91 #2

FORM TX
UNITED STATES COPYRIGHT OFFICE

REGISTRATION NUMBER

TX TXU

EFFECTIVE DATE OF REGISTRATION

Month Day Year

DO NOT WRITE ABOVE THIS LINE. IF YOU NEED MORE SPACE, USE A SEPARATE CONTINUATION SHEET.

1

TITLE OF THIS WORK ▼

Sound-to-Speech

PREVIOUS OR ALTERNATIVE TITLES ▼

PUBLICATION AS A CONTRIBUTION If this work was published as a contribution to a periodical, serial, or collection, give information about the collective work in which the contribution appeared. **Title of Collective Work ▼**

If published in a periodical or serial give: **Volume ▼** **Number ▼** **Issue Date ▼** **On Pages ▼**

2

a **NAME OF AUTHOR ▼** The Arc, formerly Association for Retarded Citizens of the United States Employer for hire of Matthew Porter

DATES OF BIRTH AND DEATH Year Born ▼ Year Died ▼

Was this contribution to the work a "work made for hire"? ☒ Yes ☐ No

AUTHOR'S NATIONALITY OR DOMICILE Name of Country OR { Citizen of ▶ ______ Domiciled in ▶ ______

WAS THIS AUTHOR'S CONTRIBUTION TO THE WORK Anonymous? ☐ Yes ☐ No Pseudonymous? ☐ Yes ☐ No If the answer to either of these questions is "Yes," see detailed instructions.

NATURE OF AUTHORSHIP Briefly describe nature of the material created by this author in which copyright is claimed. ▼

Entire Software Package

NOTE

Under the law, the "author" of a "work made for hire" is generally the employer, not the employee (see instructions). For any part of this work that was "made for hire" check "Yes" in the space provided, give the employer (or other person for whom the work was prepared) as "Author" of that part, and leave the space for dates of birth and death blank.

b **NAME OF AUTHOR ▼**

DATES OF BIRTH AND DEATH Year Born ▼ Year Died ▼

Was this contribution to the work a "work made for hire"? ☐ Yes ☐ No

AUTHOR'S NATIONALITY OR DOMICILE Name of Country OR { Citizen of ▶ ______ Domiciled in ▶ ______

WAS THIS AUTHOR'S CONTRIBUTION TO THE WORK Anonymous? ☐ Yes ☐ No Pseudonymous? ☐ Yes ☐ No If the answer to either of these questions is "Yes," see detailed instructions.

NATURE OF AUTHORSHIP Briefly describe nature of the material created by this author in which copyright is claimed. ▼

c **NAME OF AUTHOR ▼**

DATES OF BIRTH AND DEATH Year Born ▼ Year Died ▼

Was this contribution to the work a "work made for hire"? ☐ Yes ☐ No

AUTHOR'S NATIONALITY OR DOMICILE Name of Country OR { Citizen of ▶ ______ Domiciled in ▶ ______

WAS THIS AUTHOR'S CONTRIBUTION TO THE WORK Anonymous? ☐ Yes ☐ No Pseudonymous? ☐ Yes ☐ No If the answer to either of these questions is "Yes," see detailed instructions.

NATURE OF AUTHORSHIP Briefly describe nature of the material created by this author in which copyright is claimed. ▼

3

a **YEAR IN WHICH CREATION OF THIS WORK WAS COMPLETED** This information must be given in all cases. 1991 ◀ Year

b **DATE AND NATION OF FIRST PUBLICATION OF THIS PARTICULAR WORK** Complete this information ONLY if this work has been published. Month ▶ ______ Day ▶ ______ Year ▶ ______ ◀ Nation

4

COPYRIGHT CLAIMANT(S) Name and address must be given even if the claimant is the same as the author given in space 2.▼

The Arc, formerly
Association for Retarded Citizens of the United States
500 E. Border, Suite 300, Arlington, TX 76010

See instructions before completing this space.

TRANSFER If the claimant(s) named here in space 4 are different from the author(s) named in space 2, give a brief statement of how the claimant(s) obtained ownership of the copyright.▼

DO NOT WRITE HERE OFFICE USE ONLY

APPLICATION RECEIVED

ONE DEPOSIT RECEIVED

TWO DEPOSITS RECEIVED

REMITTANCE NUMBER AND DATE

BEST COPY AVAILABLE

MORE ON BACK ▶ • Complete all applicable spaces (numbers 5-11) on the reverse side of this page. • See detailed instructions. • Sign the form at line 10.

DO NOT WRITE HERE

Page 1 of ______ pages

EXAMINED BY

FORM TX

CHECKED BY

☐ CORRESPONDENCE Yes

FOR COPYRIGHT OFFICE USE ONLY

DO NOT WRITE ABOVE THIS LINE. IF YOU NEED MORE SPACE, USE A SEPARATE CONTINUATION SHEET.

5

PREVIOUS REGISTRATION Has registration for this work, or for an earlier version of this work, already been made in the Copyright Office?
☐ Yes ☒ No If your answer is "Yes," why is another registration being sought? (Check appropriate box) ▼
☐ This is the first published edition of a work previously registered in unpublished form.
☒ This is the first application submitted by this author as copyright claimant.
☐ This is a changed version of the work, as shown by space 6 on this application.
If your answer is "Yes," give: **Previous Registration Number** ▼ **Year of Registration** ▼

6

DERIVATIVE WORK OR COMPILATION Complete both space 6a & 6b for a derivative work; complete only 6b for a compilation.
a. **Preexisting Material** Identify any preexisting work or works that this work is based on or incorporates. ▼

b. **Material Added to This Work** Give a brief, general statement of the material that has been added to this work and in which copyright is claimed. ▼

See instructions before completing this space.

7

—space deleted—

8

REPRODUCTION FOR USE OF BLIND OR PHYSICALLY HANDICAPPED INDIVIDUALS A signature on this form at space 10, and a check in one of the boxes here in space 8, constitutes a non-exclusive grant of permission to the Library of Congress to reproduce and distribute solely for the blind and physically handicapped and under the conditions and limitations prescribed by the regulations of the Copyright Office: (1) copies of the work identified in space 1 of this application in Braille (or similar tactile symbols); or (2) phonorecords embodying a fixation of a reading of that work; or (3) both.
a ☐ Copies and Phonorecords b ☐ Copies Only c ☐ Phonorecords Only

See instructions.

9

DEPOSIT ACCOUNT If the registration fee is to be charged to a Deposit Account established in the Copyright Office, give name and number of Account.
Name ▼ **Account Number** ▼

CORRESPONDENCE Give name and address to which correspondence about this application should be sent. Name/Address/Apt/City/State/Zip ▼
Dr. Carrie Brown, The Arc
500 East Border, Suite 300
Arlington, TX 76010 (817)261-6003
Area Code & Telephone Number ▶

Be sure to give your daytime phone ◀ number.

10

CERTIFICATION* I, the undersigned, hereby certify that I am the
Check one ▶
☐ author
☐ other copyright claimant
☐ owner of exclusive right(s)
☒ authorized agent of The Arc
Name of author or other copyright claimant, or owner of exclusive right(s) ▲
of the work identified in this application and that the statements made by me in this application are correct to the best of my knowledge.

Typed or printed name and date ▼ If this application gives a date of publication in space 3, do not sign and submit it before that date.
Alan Abeson, Executive Director, The Arc date ▶ October 3, 1991

Handwritten signature (X) ▼

11

MAIL CERTIFICATE TO
Certificate will be mailed in window envelope

Name ▼
Dr. Carrie Brown, The Arc
Number, Street/Apartment Number ▼
500 East Border, Suite 300
City/State/ZIP ▼
Arlington, TX 76010

YOU MUST:
• Complete all necessary spaces
• Sign your application in space 10
SEND ALL 3 ELEMENTS IN THE SAME PACKAGE:
1. Application form
2. Nonrefundable $20 filing fee in check or money order payable to *Register of Copyrights*
3. Deposit material
MAIL TO:
Register of Copyrights
Library of Congress
Washington, D.C. 20559

* 17 U.S.C. § 506(e) Any person who knowingly makes a false representation of a material fact in the application for copyright registration provided for by section 409, or in any written statement filed in connection with the application, shall be fined not more than $2,500.

```
/*************************************************************************/
/* SCRIO    - Routines which directly access the video screen           */
/*                                                                       */
/*                                                                       */
/*                                                                       */
/*************************************************************************/
/*                          Modification Log                             */
/*************************************************************************/
/* Version   Date    Programmer        -----------  Description  ------------
-- */
/*                                                                       */
/* V01.00   112787  Bob Withers  Program intially complete.              */
/*                                                                       */
/*                                                                       */
/*************************************************************************/

#include <stdlib.h>
#include <stddef.h>
#include <dos.h>
#include <string.h>
#include "win.h"

#define MAXDIM(array)                    (sizeof(array) / sizeof(array[0]))

#define SCR_BYTES_PER_ROW        160
#define CGA_MODE_SEL             0x3d8
#define CGA_ENABLE               0x29
#define CGA_DISABLE              0x21

#define BIOS_VID_INT             0x10
#define BIOS_VID_SET_CRTMODE     0
#define BIOS_VID_SET_CURSORTYPE     1
#define BIOS_VID_SET_CURSORPOS          2
#define BIOS_VID_GET_CURSORPOS          3
#define BIOS_VID_SCROLL_UP       6
#define BIOS_VID_SCROLL_DOWN     7
#define BIOS_VID_WRITE_CHATTR         9
#define BIOS_VID_GET_CRTMODE     15


struct sBoxType
{
    BYTE            cUpperLeft;
    BYTE            cLowerLeft;
    BYTE            cUpperRight;
    BYTE            cLowerRight;
    BYTE            cLeft;
    BYTE            cRight;
    BYTE            cTop;
    BYTE            cBottom;
};
typedef struct sBoxType BOXTYPE;
```



```
unsigned            uScrSeg       = 0xb800;
unsigned            uCsrType      = 0x0107;
```

```
short                          nCurrActivePage = 0;
short                          nVideoCard      = VIDEO_CGA;
short                          nScrCols        = 80;


/*****************************************************************************/
/*  ScrCvtAttr          - Test for a monochrome video card and convert the
/
/*                 requested attribute to black & white as best we can    */
/*                 and still honor the callers request.                   */
/*  Parms:                                                                */
/*    nAttr     - The passed color attribute.                             */
/*                                                                        */
/*  Return Value:   A converted black & white attribute if the current   */
/*                 video mode is monochrome.                              */
/*****************************************************************************/

static short near pascal ScrCvtAttr(nAttr)
register short         nAttr;
{
    short     nRev, nClr, nBlink, nIntensity;

    if (VIDEO_MONO != nVideoCard)
     return(nAttr);
    nIntensity = nAttr & 0x40;
    nBlink     = nAttr & 0x80;
    nRev       = nAttr & 0x70;
    nClr       = nAttr & 0x07;
    if (REV_BLACK == nRev)
     nClr = WHITE;
    else
    {
     nRev = REV_WHITE;
     nClr = BLACK;
    }
    return(nRev | nClr | nBlink | nIntensity);
}


/*****************************************************************************/
/* ScrEnableVideoCGA - Test the current video equipment for a snowy CGA   */
/*                    card.  If running on a CGA enable/disable the       */
/*                    video signal based on the passed parameter which    */
/*                    MUST be one of the predefined constants CGA_ENABLE  */
/*                    or CGA_DISABLE.  If the current video equipment is  */
/*                    not a CGA, the routine returns without taking any   */
/*                    action.                                             */
/*  Parms:                                                                */
/*    nStatus              - Enable or disable the CGA video signal       */
/*                          CGA_ENABLE or CGA_DISABLE                     */
/*                                                                        */
/*  Return Value:   None                                                  */
/*****************************************************************************/

static void pascal ScrEnableVideoCGA(nStatus)
short            nStatus;
{
```

```
     if (VIDEO_CGA == nVideoCard)
      outp(CGA_MODE_SEL, nStatus);
     return;
 }


/*****************************************************************************/
/* ScrGetRectSize - This routine will calculate and return the number of     */
 /*                bytes required to store a screen image which is nWidth*/
 /*                columns by nHeight rows.                                */
 /*   Parms:                                                               */
 /*     nWidth     - Column width of the screen rectangle                  */
 /*     nHeight          - Number of rows in the screen rectangle          */
 /*                                                                        */
/* Return Value:   Size in bytes required to store the screen rectangle      */
/*****************************************************************************/

 short pascal ScrGetRectSize(nWidth, nHeight)
 short            nWidth, nHeight;
 {
     return(nWidth * nHeight * 2);
 }


/*****************************************************************************/
 /*   ScrClearRect - This routine will clear a screen rectangle to the      */
 /*              color attribute passed.                                    */
 /*   Parms:                                                                */
 /*     nRow     - Row of the screen rectangle                              */
 /*     nCol     - Column of the screen rectangle                           */
 /*     nWidth   - Width in columns of the screen rectangle                 */
 /*     nHeight        - Number of rows in the screen rectangle             */
 /*     nAttr    - Color attribute used to clear screen rectangle           */
 /*                                                                         */
 /*   Return Value:  None                                                   */
/*****************************************************************************/

 void pascal ScrClearRect(nRow, nCol, nWidth, nHeight, nAttr)
 short                          nRow, nCol, nWidth, nHeight, nAttr;
 {
     auto     union REGS     r;

     nAttr  = ScrCvtAttr(nAttr);
     r.h.ah = (BYTE) BIOS_VID_SCROLL_UP;
     r.h.al = 0;
     r.h.bh = (BYTE) nAttr;
     r.h.ch = (BYTE) (nRow - 1);
     r.h.cl = (BYTE) (nCol - 1);
     r.h.dh = (BYTE) (nRow + nHeight - 2);
     r.h.dl = (BYTE) (nCol + nWidth - 2);
     int86(BIOS_VID_INT, &r, &r);
     return;
 }


/*****************************************************************************/
```

```
/* ScrSaveRect - This routine will save a screen rectangle in a caller    */
/*               supplied buffer area.  nRow, nCol define the row and     */
/*               column of the upper left corner of the rectangle.        */
/*  Parms:                                                                */
/*    nRow   - Row of the screen rectangle                                */
/*    nCol   - Column of the screen rectangle                             */
/*    nWidth - Width in columns of the screen rectangle                   */
/*    nHeight       - Number of rows in the screen rectangle              */
/*    pBuf   - Buffer used to store the saved screen rectangle            */
/*                                                                        */
/*  Return Value:  None                                                   */
/************************************************************************/

void pascal ScrSaveRect(nRow, nCol, nWidth, nHeight, pBuf)
short            nRow, nCol, nWidth, nHeight;
char   *pBuf;
{
    register unsigned            uNumRows;
    register unsigned            uColLen;
    auto     unsigned            uScrOfs;
    auto     unsigned            uBufSeg, uBufOfs;
    auto     char far           *fpBuf;

    uColLen = nWidth * 2;
    uScrOfs = ((nRow - 1) * SCR_BYTES_PER_ROW) + (nCol - 1) * 2;
    fpBuf   = (char far *) pBuf;
    uBufSeg = FP_SEG(fpBuf);
    uBufOfs = FP_OFF(fpBuf);
    ScrEnableVideoCGA(CGA_DISABLE);
    for (uNumRows = nHeight; uNumRows > 0; --uNumRows)
    {
     movedata(uScrSeg, uScrOfs, uBufSeg, uBufOfs, uColLen);
     uScrOfs += SCR_BYTES_PER_ROW;
     uBufOfs += uColLen;
    }
    ScrEnableVideoCGA(CGA_ENABLE);
    return;
}


/************************************************************************/
/*  ScrRestoreRect - This routine will restore a screen rectangle from    */
/*                   a previously saved caller buffer.  nRow and nCol      */
/*                   define the upper left corner of the rectangle on      */
/*                   the screen and are not required to be the same        */
/*                   coordinates used in the save call.  nWidth and        */
/*                   nHeight should remain unchanged from the save call    */
/*                   but are not required to do so.                        */
/*  Parms:                                                                */
/*    nRow   - Row of the screen rectangle                                */
/*    nCol   - Column of the screen rectangle                             */
/*    nWidth - Width in columns of the screen rectangle                   */
/*    nHeight       - Number of rows in the screen rectangle              */
/*    pBuf   - Buffer used to restore the saved screen rectangle          */
/*                                                                        */
/*  Return Value:  None                                                   */
```

```
/*****************************************************************************/

void pascal ScrRestoreRect(nRow, nCol, nWidth, nHeight, pBuf)
short           nRow, nCol, nWidth, nHeight;
char   *pBuf;
{
    register unsigned           uNumRows;
    register unsigned           uColLen;
    auto     unsigned           uScrOfs;
    auto     unsigned           uBufSeg, uBufOfs;
    auto     char far          *fpBuf;

    uColLen = nWidth * 2;
    uScrOfs = ((nRow - 1) * SCR_BYTES_PER_ROW) + (nCol - 1) * 2;
    fpBuf   = (char far *) pBuf;
    uBufSeg = FP_SEG(fpBuf);
    uBufOfs = FP_OFF(fpBuf);
    ScrEnableVideoCGA(CGA_DISABLE);
    for (uNumRows = nHeight; uNumRows > 0; --uNumRows)
    {
     movedata(uBufSeg, uBufOfs, uScrSeg, uScrOfs, uColLen);
     uScrOfs += SCR_BYTES_PER_ROW;
     uBufOfs += uColLen;
    }
    ScrEnableVideoCGA(CGA_ENABLE);
    return;
}


/*****************************************************************************/
/* ScrScrollRectUp  - Scrolls a screen rectangle up the requested number */
/*                    of lines.                                        */
/*  Parms:                                                             */
/*    nRow   - Row of the screen rectangle                             */
/*    nCol   - Column of the screen rectangle                          */
/*    nWidth - Width in columns of the screen rectangle                */
/*    nHeight        - Number of rows in the screen rectangle          */
/*    nNoRows        - Number of rows to scroll                        */
/*    nAttr  - Color attribute to fill blank line on bottom            */
/*                                                                     */
/*  Return Value:  None                                                */
/*****************************************************************************/

void pascal ScrScrollRectUp(nRow, nCol, nWidth, nHeight, nNoRows, nAttr)
short          nRow, nCol, nWidth, nHeight, nNoRows, nAttr;
{
    auto     union REGS      r;

    nAttr  = ScrCvtAttr(nAttr);
    r.h.ah = BIOS_VID_SCROLL_UP;
    r.h.al = (BYTE) nNoRows;
    r.h.bh = (BYTE) nAttr;
    r.h.ch = (BYTE) (nRow - 1);
    r.h.cl = (BYTE) (nCol - 1);
    r.h.dh = (BYTE) (nRow + nHeight - 2);
    r.h.dl = (BYTE) (nCol + nWidth - 2);
```

```
    int86(BIOS_VID_INT, &r, &r);
    return;
}


/****************************************************************************/
/* ScrScrollRectDown - Scrolls a screen rectangle up the requested number */
/*                        of lines.                                        */
/*  Parms:                                                                 */
/*    nRow    - Row of the screen rectangle                                */
/*    nCol    - Column of the screen rectangle                             */
/*    nWidth - Width in columns of the screen rectangle                    */
/*    nHeight         - Number of rows in the screen rectangle             */
/*    nNoRows         - Number of rows to scroll                           */
/*    nAttr   - Color attribute to fill blank lines on top                 */
/*                                                                         */
/*  Return Value:  None                                                    */
/****************************************************************************/

void pascal ScrScrollRectDown(nRow, nCol, nWidth, nHeight, nNoRows, nAttr)
short         nRow, nCol, nWidth, nHeight, nNoRows, nAttr;
{
    auto      union REGS      r;

    nAttr   = ScrCvtAttr(nAttr);
    r.h.ah = BIOS_VID_SCROLL_DOWN;
    r.h.al = (BYTE) nNoRows;
    r.h.bh = (BYTE) nAttr;
    r.h.ch = (BYTE) (nRow - 1);
    r.h.cl = (BYTE) (nCol - 1);
    r.h.dh = (BYTE) (nRow + nHeight - 2);
    r.h.dl = (BYTE) (nCol + nWidth - 2);
    int86(BIOS_VID_INT, &r, &r);
    return;
}


/****************************************************************************/
/* ScrSetCursorPos - This routine will position the cursor to an absolute */
/*                       screen coordinate using the BIOS video services.  */
/*  Parms:                                                                 */
/*    nRow         - Absolute screen row                                   */
/*    nCol         - Absolute screen column                                */
/*                                                                         */
/*  Return Value       None                                                */
/****************************************************************************/

void pascal ScrSetCursorPos(nRow, nCol)
short             nRow, nCol;
{
    auto      union REGS     r;

    r.h.ah = BIOS_VID_SET_CURSORPOS;
    r.h.dh = (BYTE) (nRow - 1);
    r.h.dl = (BYTE) (nCol - 1);
    r.h.bh = (BYTE) nCurrActivePage;
```

```
        int86(BIOS_VID_INT, &r, &r);
        return;
    }


/*************************************************************************/
  /*  ScrGetCursorPos - This routine will return the current absolute    */
  /*                    cursor position.                                  */
  /*  Parms:                                                              */
  /*    nRow        - Pointer to location to save current row             */
  /*    nCol        - Pointer to location to save current column          */
  /*                                                                      */
  /*  Return Value:      None                                             */
/*************************************************************************/

  void pascal ScrGetCursorPos(nRow, nCol)
  short             *nRow, *nCol;
  {
      auto      union REGS      r;

      r.h.ah = BIOS_VID_GET_CURSORPOS;
      r.h.bh = (BYTE) nCurrActivePage;
      int86(BIOS_VID_INT, &r, &r);
      *nRow   = r.h.dh + 1;
      *nCol   = r.h.dl + 1;
      return;
  }


/*************************************************************************/
  /*  ScrCusrosOn      - Enables the screen cursor.                       */
  /*                                                                      */
  /*  Parms:          None                                                */
  /*                                                                      */
  /*  Return Value:      None                                             */
/*************************************************************************/

  void pascal ScrCursorOn()
  {
      auto      union REGS      r;

      r.h.ah = BIOS_VID_SET_CURSORTYPE;
      r.x.cx = uCsrType;
      int86(BIOS_VID_INT, &r, &r);
      return;
  }


/*************************************************************************/
  /*  ScrCusrosOff     - Disables the screen cursor.                      */
  /*                                                                      */
  /*  Parms:          None                                                */
  /*                                                                      */
  /*  Return Value:      None                                             */
/*************************************************************************/
```

```
void pascal ScrCursorOff()
{
    auto     union REGS     r;

    r.h.ah = BIOS_VID_SET_CURSORTYPE;
    r.x.cx = 0x0f00;
    int86(BIOS_VID_INT, &r, &r);
    return;
}


/*****************************************************************************/
/* ScrTextOut - This function uses the BIOS write character and attribute */
/*              service routine to display a string within a window.  The */
/*              passed nCount is used to limit a string from overflowing  */
/*              a window boundry.                                          */
/*   Parms:                                                                */
/*     pStr     - Pointer to the string to be displayed                    */
/*     nAttr    - Color attribute used to display string                   */
/*     nCount   - Maximum number of characters to dispalay                 */
/*                                                                         */
/*   Return Value: None                                                    */
/*****************************************************************************/

void pascal ScrTextOut(pStr, nAttr, nCount)
register char       *pStr;
short                nAttr, nCount;
{
    auto     short       nRow, nCol;
    auto     union REGS     r, r1;

    ScrGetCursorPos(&nRow, &nCol);
    nAttr  = ScrCvtAttr(nAttr);
    r.h.ah = BIOS_VID_WRITE_CHATTR;
    r.h.bh = (BYTE) nCurrActivePage;
    r.h.bl = (BYTE) nAttr;
    r.x.cx = 1;
    while (*pStr && nCount-- > 0)
    {
     ScrSetCursorPos(nRow, nCol++);
     r.h.al = *pStr++;
     int86(BIOS_VID_INT, &r, &r1);
    }
    return;
}


/*****************************************************************************/
/*   ScrDrawRect - This routine is used to draw borders around a screen   */
/*                 window.  The passed parameters define the rectangle    */
/*                 being used by the window as well as the border color   */
/*                 and type.                                              */
/*   Parms:                                                               */
/*     nRow   - Top row of screen border                                  */
/*     nCol   - Left column of screen border                              */
/*     nWidth - Column width of the window                                */
```

```
/*    nHeight        - Number of rows in the window                          */
/*    nColor - Color attribute for the window border                         */
/*    nType  - Type of border to be displayed                                */
/*                                                                           */
/*  Return Value: None                                                       */
/*****************************************************************************/

void pascal ScrDrawRect(nRow, nCol, nWidth, nHeight, nColor, nType)
short           nRow, nCol, nWidth, nHeight, nColor, nType;
{
    register short      i;
    auto     union REGS    r, r1;
    static   BOXTYPE    BoxTypes[] =
    {
     {  32,  32,  32,  32,  32,  32,  32,  32 }, /* NO_BOX               */
     { 213, 212, 184, 190, 179, 179, 205, 205 }, /* DBL_LINE_TOP_BOTTOM */
     { 214, 211, 183, 189, 186, 186, 196, 196 }, /* DBL_LINE_SIDES      */
     { 201, 200, 187, 188, 186, 186, 205, 205 }, /* DBL_LINE_ALL_SIDES  */
     { 218, 192, 191, 217, 179, 179, 196, 196 }, /* SNGL_LINE_ALL_SIDES */
     { 219, 219, 219, 219, 219, 219, 223, 220 }  /* GRAPHIC BOX      */
    };

    if (nType < 0 || nType >= MAXDIM(BoxTypes))
     return;
    if (nWidth < 2 || nHeight < 2)
     return;
    nColor = ScrCvtAttr(nColor);

    /* Draw upper left corner */
    ScrSetCursorPos(nRow, nCol);
    r.h.ah = (BYTE) BIOS_VID_WRITE_CHATTR;
    r.h.al = (BYTE) BoxTypes[nType].cUpperLeft;
    r.h.bh = (BYTE) nCurrActivePage;
    r.h.bl = (BYTE) nColor;
    r.x.cx = 1;
    int86(BIOS_VID_INT, &r, &r1);

    /* Draw upper right corner */
    ScrSetCursorPos(nRow, nCol + nWidth - 1);
    r.h.al = (BYTE) BoxTypes[nType].cUpperRight;
    int86(BIOS_VID_INT, &r, &r1);

    /* Draw lower left corner */
    ScrSetCursorPos(nRow + nHeight - 1, nCol);
    r.h.al = (BYTE) BoxTypes[nType].cLowerLeft;
    int86(BIOS_VID_INT, &r, &r1);

    /* Draw lower right corner */
    ScrSetCursorPos(nRow + nHeight - 1, nCol + nWidth - 1);
    r.h.al = (BYTE) BoxTypes[nType].cLowerRight;
    int86(BIOS_VID_INT, &r, &r1);

    if (nHeight > 2)
    {
     /* Draw left side line */
     r.h.al = (BYTE) BoxTypes[nType].cLeft;
```

```
        for (i = 1; i <= nHeight - 2; ++i)
        {
            ScrSetCursorPos(nRow + i, nCol);
            int86(BIOS_VID_INT, &r, &r1);
        }

        /* Draw right side line */
        r.h.al = (BYTE) BoxTypes[nType].cRight;
        for (i = 1; i <= nHeight - 2; ++i)
        {
            ScrSetCursorPos(nRow + i, nCol + nWidth - 1);
            int86(BIOS_VID_INT, &r, &r1);
        }
       }

       if (nWidth > 2)
       {
        /* Draw top line */
        ScrSetCursorPos(nRow, nCol + 1);
        r.h.al = (BYTE) BoxTypes[nType].cTop;
        r.x.cx = nWidth - 2;
        int86(BIOS_VID_INT, &r, &r1);

        /* Draw bottom line */
        ScrSetCursorPos(nRow + nHeight - 1, nCol + 1);
        r.h.al = BoxTypes[nType].cBottom;
        int86(BIOS_VID_INT, &r, &r1);
       }

       return;
   }


/*********************************************************************/
 /*  ScrInitialize - Determine type of video card and init global data.  */
 /*                                                                     */
 /*  Parms:        None                                                 */
 /*                                                                     */
 /*  Return Value:   None                                               */
/*********************************************************************/

 void pascal ScrInitialize()
 {
     auto     union REGS     r;

     r.h.ah = BIOS_VID_GET_CURSORPOS;
     r.h.bh = (BYTE) nCurrActivePage;
     int86(BIOS_VID_INT, &r, &r);
     uCsrType = r.x.cx;

     r.h.ah = BIOS_VID_GET_CRTMODE;
     int86(BIOS_VID_INT, &r, &r);
     nScrCols      = r.h.ah;
     nCurrActivePage = r.h.bh;
      if (7 == r.h.al)
     {
```

```
         uScrSeg    = 0xb000;
         nVideoCard = VIDEO_MONO;
         return;
        }
        r.h.ah = BIOS_VID_SET_CRTMODE;
        r.h.al = 3;
         int86(BIOS_VID_INT, &r, &r);
        uScrSeg    = 0xb800;
        nVideoCard = VIDEO_CGA;
        return;
    }
```

```
#include<conio.h>
#include<string.h>
#include<win.h>
#include<keyboard.h>

#define MAX_STR_LEN  80

/*********************************************************************/
/* FUNCTION   get_key                                                  */
/* REMARKS    waits for a keystroke then gets a keystroke scan code */
/*            from the keyboard buffer                                 */
/* INPUT                                                               */
/* OUTPUT     the value of the key hit, if the key was a function   */
/*            key of any kind, then the negative of the scan code is*/
/*            returned                                                 */
/* CALLED BY  get_str                                                  */
/* CALLS      msg_of_day, update_clock                              */
/*********************************************************************/

int get_key(void)
{
int ch;

 ch = getch();
 if (ch == 0)
   ch = -getch();

 return ch;

}/*get_key*/

/*********************************************************************/
/* FUNCTION   get_str                                                  */
/* REMARKS    Gets a string from the keyboard.  To exit either press*/
/*            RETURN, or ESC.  This function supports the backspace.*/
/* INPUT      int lenstr   : - length of the string to input         */
/*                           - if lenstr > 80 then lenstr = 80     */
/*         int x & y    : x and y screen locations to print the   */
/*                         string as it is being typed in          */
/*            int *last_key: the last keystroke                       */
/* OUTPUT     the string typed in                                   */
/* CALLED BY                                                        */
/* CALLS      get_key                                                  */
/*********************************************************************/
char *get_str(HWND wh, int lenstr, int x, int y, int *last_key)
{
char str[MAX_STR_LEN]="";
char ch=0;
short tx=0, ty=0;
int num_letters=0;

 if (lenstr > MAX_STR_LEN)
  lenstr = MAX_STR_LEN;

for (tx=x;tx<=(x+lenstr-1);tx++)   /* clear out the space about to be typed in*/
   {
```

```
      WinSetCursorPos(wh, y, tx);
      cprintf(" ");
    }/*for*/

   WinSetCursorPos(wh, y, x);
   while (ch != UP && ch != DOWN && ch != ESC && ch != RETURN)
    {
     ch = get_key();
     *last_key = ch;
     if (ch == BACKSPACE || ch >= SPACEBAR && ch <= TILDA)
       {
        if (ch == BACKSPACE)
         {
       if (num_letters-- < 1)
          num_letters = 0;          /* backup internal cursor 1 element*/
       str[num_letters] = 0;         /* assign end of string   */

          ScrGetCursorPos(&ty, &tx);
          if (tx == 1)                  /* Handles backing up past the */
         {                           /* beginning of a line.*/
          tx = 81;
          ty--;
         }/*if*/

          ScrSetCursorPos(ty, tx-1);
          if (strlen(str))
            cprintf(" ");
       }/*if*/
         else
       if (num_letters < lenstr && ch>= SPACEBAR && ch <=TILDA)
         {
            str[num_letters++] = ch;
            str[num_letters] = 0;
         }/*if*/
       }/*if*/
     WinSetCursorPos(wh, y, x);
     cprintf("%s", str);
     }/*while*/

  if (ch != ESC && ch != DOWN && ch != UP)
    return str;
   else
    return "";

  }/*get_str*/
```

```
#define DEBUG 1

#include<conio.h>
#include<string.h>
#include<alloc.h>
#include<process.h>
#include<dir.h>
#include<stdio.h>
#include"win.h"
#include"pointsht.h"
#include"keyboard.h"
#include"student.h"
#define BLANK_STR_LEN_8 "        "

char *program_root_dir; /* sts package root directory */

void clear_student_space(void);
int  put_message(char *);
void main_menu(void);
int  get_filenames(char *, char *);
void title_screen(void);

#define ADD_STUDENT    2
#define SELECT_STUDENT 3
#define DELETE_STUDENT 0
#define EDIT_STUDENT   1
#define MAX_STUDENTS   20     /* max number of students*/
#define NAME_LEN 21
  char  *name;
  char  *age;
  char  *diagnosis;
  char  *person_1;
  char  *person_2;
  char  *location;
 };

struct student_type student;

/**********************************************************************/
/* FUNCTION   free_all                                                */
/* REMARKS    frees all allocated memory space allocated by malloc */
/* INPUT      int num_ts - number of main menu items                  */
/* OUTPUT                                                             */
/* CALLED BY  main                                                    */
/* CALLS                                                              */
/**********************************************************************/

void free_all(int num_ts
```

```
#define DEBUG 1

#include<conio.h>
#include<string.h>
#include<alloc.h>
#include<process.h>
#include<dir.h>
#include<stdio.h>
#include"win.h"
#include"pointsht.h"
#include"keyboard.h"
#include"student.h"
#define BLANK_STR_LEN_8 "        "

char *program_root_dir; /* sts package root directory */

void clear_student_space(void);
int  put_message(char *);
void main_menu(void);
int  get_filenames(char *, char *);
void title_screen(void);

#define ADD_STUDENT    2
#define SELECT_STUDENT 3
#define DELETE_STUDENT 0
#define EDIT_STUDENT   1
#define MAX_STUDENTS   20     /* max number of students*/
#define NAME_LEN 21
#define AGE_LEN 3
#define DIAG_LEN 24
#define ADD_ST  "Add Student"
#define SELECT_ST  "Select Student"
#define DEL_ST  "Delete Student"
#define EDIT_ST "Edit Student Description"
#define EXIT_ST "Exit"

short message_color = WHITE|REV_BLUE|HI_INTENSITY;
short warning_color = WHITE|REV_RED|HI_INTENSITY;
short win_color = WHITE|REV_GREEN|HI_INTENSITY;
short bdr_color = WHITE|REV_GREEN|HI_INTENSITY;
short reverse_color = WHITE|HI_INTENSITY;
short EXPLODE = TRUE;

FILE *fopen();

struct text_type *text1;
struct text_type *name_list;

struct student_type{
  char  *name;
  char  *age;
  char  *diagnosis;
  char  *person_1;
  char  *person_2;
  char  *location;
 };
```

```
struct student_type student;

/*********************************************************************/
/* FUNCTION    free_all                                              */
/* REMARKS     frees all allocated memory space allocated by malloc */
/* INPUT       int num_ts - number of main menu items               */
/* OUTPUT                                                            */
/* CALLED BY   main                                                  */
/* CALLS                                                             */
/*********************************************************************/

void free_all(int num_ts)
{
int c;

 flushall();
 fcloseall();
 for (c=0;c<num_ts;c++)
  free(text1[c].str);

 for (c=0;c<MAX_STUDENTS;c++)
   free(name_list[c].str);
 free(student.name);
 free(student.age);
 free(student.diagnosis);
 free(student.person_1);
 free(student.person_2);
 free(student.location);
 free(text1);
 free(name_list);
}

/*********************************************************************/
/* FUNCTION    wtext_out                                             */
/* REMARKS     Prints text to a window at a specified row, col and  */
/*             color                                                 */
/* INPUT       HWND wh - window handle to print to                   */
/*             int col - column of window to print at                */
/*             int row - row of window to print at                   */
/*             char *str - string to print                           */
/*             int color - color attributes for the string to print */
/* OUTPUT                                                            */
/* CALLED BY   open_and_print_student_window, display_st_info_str,  */
/*             create_add_student_window, add_or_edit_student,      */
/*             delete_or_edit_student                                */
/* CALLS                                                             */
/*********************************************************************/

void wtext_out(HWND wh, int col, int row, char *str, int color)
{
 WinSetCursorPos(wh, row, col);
 WinTextOut(wh, str, color);
}

/*********************************************************************/
```

```
/* FUNCTION   put_warning_msg                                              */
/* REMARKS    Opens a red and white window at the center of the            */
/*            screen, and prints a message                                 */
/* INPUT      char *str - string or message to print                       */
/* OUTPUT                                                                  */
/* CALLED BY  save_student_info                                            */
/* CALLS      get_key                                                      */
/***************************************************************************/

void put_warning_msg(char *str)
{
 HWND wh;

wh = WinCreateWindow(11, 10, 60, 3, warning_color, DBL_LINE_SIDES, warning_color, EXPLODE);
 WinCenterText(wh, 1, str, warning_color);
 get_key();
 WinDestroyWindow(wh);
}

/***************************************************************************/
/* FUNCTION   assign_menu_attribs                                          */
/* REMARKS    assigns the menu attributes for the main menu to             */
/*            allow for point and shoot routine                            */
/* INPUT                                                                   */
/* OUTPUT                                                                  */
/* CALLED BY  main                                                         */
/* CALLS      allocate_text_object                                         */
/***************************************************************************/


void assign_menu_attribs(void)
{
int first_row = 2;
int lcol = 2;

 allocate_text_object(text1, 0, ADD_ST, lcol, first_row);
 allocate_text_object(text1, 1, DEL_ST, lcol, first_row+2);
 allocate_text_object(text1, 2, EDIT_ST, lcol, first_row+4);
 allocate_text_object(text1, 3, SELECT_ST, lcol, first_row+6);
 allocate_text_object(text1, 4, EXIT_ST, lcol, first_row+10);

}


/***************************************************************************/
/* FUNCTION   open_and_print_student_window                                */
/* REMARKS    opens and prints the main window of the student              */
/*            program                                                      */
/* INPUT      HWND *wh - pointer to the window handle                      */
/*            int num_ts - number of menu items                            */
/* OUTPUT                                                                  */
/* CALLED BY  main                                                         */
/* CALLS      wtext_out                                                    */
/***************************************************************************/

void open_and_print_student_window(HWND *wh, int num_ts)
```

```
{
int c;

*wh = WinCreateWindow(7, 15, 50, 14, win_color, DBL_LINE_ALL_SIDES, bdr_color, EXPLODE);
  WinCenterText(*wh, 0, "Student", win_color);
  for (c=0;c<num_ts;c++)
   wtext_out(*wh, text1[c].col, text1[c].row, text1[c].str, win_color);

}


/***********************************************************************/
/* FUNCTION   create_add_student_window                                */
/* REMARKS    creates and prints the text1 for the attribute input     */
/* INPUT      HWND *st_wh - pointer to the student window              */
/*            int lcol - left column the attribute names will be       */
/*            printed at                                               */
/* OUTPUT                                                                   */
/* CALLED BY  add_or_edit_student                                      */
/* CALLS      wtext_out                                                     */
/***********************************************************************/

void create_add_student_window(int OPERATION, HWND *st_wh, int lcol)
{

*st_wh = WinCreateWindow(7, 15, 50, 18, win_color, DBL_LINE_SIDES, bdr_color, EXPLODE);
  switch(OPERATION)
   {
    case ADD_STUDENT : WinCenterText(*st_wh, 0, "Add Student", win_color);
                       break;
    case EDIT_STUDENT: WinCenterText(*st_wh, 0, "Edit Student", win_color);
                       break;
   }/*switch*/

  wtext_out(*st_wh, lcol, 2, "Name:[", win_color);
  wtext_out(*st_wh, lcol+5+NAME_LEN, 2, "]", win_color);
  wtext_out(*st_wh, lcol, 4, "Age:[", win_color);
  wtext_out(*st_wh, lcol+4+AGE_LEN, 4, "]", win_color);
  wtext_out(*st_wh, lcol, 6, "Diagnosis:[", win_color);
  wtext_out(*st_wh, lcol+10+DIAG_LEN, 6, "]", win_color);
  wtext_out(*st_wh, lcol, 8, "Person 1:[", win_color);
  wtext_out(*st_wh, lcol+9+NAME_LEN, 8, "]", win_color);
  wtext_out(*st_wh, lcol, 10, "Person 2:[", win_color);
  wtext_out(*st_wh, lcol+9+NAME_LEN, 10, "]", win_color);
  wtext_out(*st_wh, lcol, 12, "Location :[", win_color);
  wtext_out(*st_wh, lcol+DIAG_LEN+10, 12,  "]", win_color);
  wtext_out(*st_wh, lcol, 14, "Exit and Save", win_color);
  wtext_out(*st_wh, lcol, 16, "Select Student", win_color);


  /*---------------------clear out new space ---------------*/
  clear_student_space();

}
/***********************************************************************/
/* FUNCTION   display_st_info_str                                      */
```

```
/* REMARKS     displays the individual attributes of each student  */
/* INPUT       HWND wh - window handle to print to                 */
/*             char *str - character string to print               */
/*             char *blank - blank string                          */
/*             int col - column to print at                        */
/*             int row - row to print at                           */
/* OUTPUT                                                             */
/* CALLED BY   display_student_info                                */
/* CALLS       wtext_out                                              */
/*******************************************************************/

void display_st_info_str(HWND wh, char *str, char *blank, int col, int row)
{
 if (strcmp(str, NULL) && strcmp(str, "-"))
   wtext_out(wh, col, row, str, win_color);
  else
   wtext_out(wh, col, row, blank,  win_color);

}

/*******************************************************************/
/* FUNCTION    display_student_info                                */
/* REMARKS     center for printing student attributes              */
/* INPUT       HWND wh - window to print to                        */
/*             int lcol - column to print at                       */
/* OUTPUT                                                             */
/* CALLED BY   add_or_edit_student                                 */
/* CALLS       display_st_info_str                                 */
/*******************************************************************/


void display_student_info(HWND wh, int lcol)
{
 display_st_info_str(wh, student.name, BLANK_STR_LEN_8, lcol+6, 2);
 display_st_info_str(wh, student.age, "  ", lcol+5, 4);
 display_st_info_str(wh, student.diagnosis, BLANK_STR_LEN_8, lcol+11, 6);
 display_st_info_str(wh, student.person_1, BLANK_STR_LEN_8, lcol+10, 8);
 display_st_info_str(wh, student.person_2, BLANK_STR_LEN_8, lcol+10, 10);
 display_st_info_str(wh, student.location, "           ", lcol+11, 12);


 wtext_out(wh, lcol, 14, "Exit and Save", win_color);
 wtext_out(wh, lcol, 16, "Select Student", win_color);


}
/*******************************************************************/
/* FUNCTION    write_str2file                                      */
/* REMARKS     If the string exists then the string is written to  */
/*             a file, else a "-" is written.                      */
/* INPUT       FILE *fp - file to write to                         */
/*             char *str - string to write                         */
/* OUTPUT                                                             */
/* CALLED BY   update_student_list_file, save_student_info         */
/* CALLS                                                              */
```

```
/**********************************************************************/

void write_str2file(FILE *fp, char *str)
{
if (str && strlen(str)>0)
  {
   fprintf(fp, "%s\n", str);
  }/*if*/
 else
  {
   fprintf(fp, "-\n");
  }/*else*/

}
/**********************************************************************/
/* FUNCTION   update_student_list_file                                */
/* REMARKS    updates the student list by adding the new name         */
/* INPUT      FILE *fp - file handle for the student list file        */
/* OUTPUT                                                                  */
/* CALLED BY  save_student_info                                       */
/* CALLS      write_str2file,                                         */
/**********************************************************************/


void update_student_list_file(void)
{
FILE *fp1;
char str[10]="";
int FOUND = FALSE;

  /* check to ensure that the name does not exist in the file */
  fp1 = fopen("studlist.dat", "rt");
  if (fp1)
   {
     while(!feof(fp1) && strcmp(str, ""))
        {
          fgets(str, 10, fp1);
          str[strlen(str)-1]=0;
          if(!strcmp(str, student.name))
              FOUND = TRUE;
        }/*while*/
   }/*if*/
  fflush(fp1);
  fclose(fp1);


  if (!FOUND)
   {
     fp1 = fopen("studlist.dat", "at");
     if (fp1)
      {
       write_str2file(fp1, student.name);

      }/*if*/
     fflush(fp1);
     fclose(fp1);
```

```
    }/*if*/
  }
  /*************************************************************************/
  /* FUNCTION   save_student_info                                          */
  /* REMARKS    saves the student attributes to a file                     */
  /* INPUT                                                                 */
  /* OUTPUT                                                                */
  /* CALLED BY  add_or_edit_student                                        */
  /* CALLS      put_warinng_msg, write_str2file                            */
  /*************************************************************************/

  void save_student_info(int OPERATION)
  {
    FILE *fp;
    char buf[100];

    strcpy(buf, program_root_dir);

    if (mkdir(student.name) && OPERATION == ADD_STUDENT)
      put_warning_msg("You need a unique name!");
    else
     {
      strcat(buf, "\\");
      strcat(buf, student.name);
      chdir(buf);
      fp = fopen("students.dat", "wt");
      if (fp)
      {
       write_str2file(fp, student.name);
       write_str2file(fp, student.age);
       write_str2file(fp, student.diagnosis);
       write_str2file(fp, student.person_1);
       write_str2file(fp, student.person_2);
       write_str2file(fp, student.location);
      }/*if*/

      fflush(fp);
      fclose(fp);
      chdir(program_root_dir);
      if (OPERATION == ADD_STUDENT)
        update_student_list_file();
     }/*else*/

  }

  /*************************************************************************/
  /* FUNCTION   clear_student_space                                        */
  /* REMARKS    sets all student attributes to null                        */
  /* INPUT                                                                 */
  /* OUTPUT                                                                */
  /* CALLED BY  create_add_student_window, add_or_edit_student             */
  /* CALLS                                                                 */
  /*************************************************************************/

  void clear_student_space(void)
  {
```

```
 strcpy(student.name, "-");
 strcpy(student.age, "-");
 strcpy(student.diagnosis, "-");
 strcpy(student.person_1, "-");
 strcpy(student.person_2, "-");
 strcpy(student.location, "-");
}

/*************************************************************************/
/* FUNCTION    read_str_from_file                                        */
/* REMARKS     reads a string from a file                                */
/* INPUT       FILE *fp - file to read from                              */
/*             char *str - pointer to string to return                   */
/* OUTPUT                                                                */
/* CALLED BY   read_student_data                                         */
/* CALLS                                                                 */
/*************************************************************************/

void read_str_from_file(FILE *fp, char *str)
{
 fgets(str, NAME_LEN+1, fp);
 str[strlen(str)-1] = 0;

}

/*************************************************************************/
/* FUNCTION    check4null                                                */
/* REMARKS     checks to see if the string is blank                      */
/* INPUT       char *str - string to check                               */
/* OUTPUT                                                                */
/* CALLED BY   read_student_data                                         */
/* CALLS                                                                 */
/*************************************************************************/

void check4null(char *str)
{
 if (!strcmp(str, "-"))
  strcpy(str, NULL);
}

/*************************************************************************/
/* FUNCTION    read_student_data                                         */
/* REMARKS     reads in the student data from the file                   */
/* INPUT       char *name - the students name/directory to read from*/
/* OUTPUT                                                                */
/* CALLED BY   add_or_edit_student                                       */
/* CALLS       read_str_from_file, read_student_data                     */
/*************************************************************************/

void read_student_data(char *name)
{
 FILE *fp;
 char str[NAME_LEN]="", buf[100];

 strcpy(buf, program_root_dir);
 strcat(buf, "\\");
```

```
    strcat(buf, name);

    chdir(buf);
    fp = fopen("students.dat", "rt");

    read_str_from_file(fp, str);
    strcpy(student.name, str);
    check4null(student.name);

    read_str_from_file(fp, str);
    strcpy(student.age, str);
    check4null(student.age);

    read_str_from_file(fp, str);
    strcpy(student.diagnosis, str);
    check4null(student.diagnosis);

    read_str_from_file(fp, str);
    strcpy(student.person_1, str);
    check4null(student.person_1);

    read_str_from_file(fp, str);
    strcpy(student.person_2, str);
    check4null(student.person_2);

    read_str_from_file(fp, str);
    strcpy(student.location, str);
    check4null(student.location);

    fflush(fp);
    fclose(fp);
    chdir(program_root_dir);
   }

   /**********************************************************************/
   /* FUNCTION   add_or_edit_student                                     */
   /* REMARKS    control center for adding a student to the STS system*/
   /* INPUT      int OPERATION - ADD or EDIT                             */
   /*            char *name - name of student to edit                    */
   /* OUTPUT                                                                  */
   /* CALLED BY  main                                                    */
   /* CALLS      create_add_student_window, clear_student_space,         */
   /*            display_student_info, wtext_out, read_student_data      */
   /**********************************************************************/

   char* add_or_edit_student(int OPERATION, char *name)
   {
    HWND st_wh;
    int  lcol = 2, num_items = 7;
    char *str;
    int  item = 0;
    int  FINISHED = FALSE;
    int  last_key = 0;
    int rc = 0;

    str = (char *)malloc(sizeof(char) * NAME_LEN);
```

```
    create_add_student_window(OPERATION, &st_wh, lcol);
    switch(OPERATION)
     {
       case ADD_STUDENT:  clear_student_space();
                          break;

       case EDIT_STUDENT: read_student_data(name);
                          break;

     }/*switch*/

    textattr(win_color);
    while(!FINISHED)
     {
     display_student_info(st_wh, lcol);
     switch(item)
      {
        case 0 : if (OPERATION != EDIT_STUDENT)
                       {
                              strcpy(str, get_str(st_wh, NAME_LEN-1, lcol+6,
, &last_key));
                              if (strcmp(str, NULL))
                                      strcpy(student.name, str);
                             break;
                       }/*else*/
                       else
                              item=1;

       case 1 : strcpy(str, get_str(st_wh, AGE_LEN-1, lcol+5, 4, &last_key));
                if (strcmp(str, NULL))
                   strcpy(student.age, str);
                break;
       case 2 : strcpy(str, get_str(st_wh, DIAG_LEN-1, lcol+11, 6, &last_key));
                if (strcmp(str, NULL))
                   strcpy(student.diagnosis, str);
                break;
       case 3 : strcpy(str, get_str(st_wh, NAME_LEN-1, lcol+10, 8, &last_key));
                if (strcmp(str, NULL))
                   strcpy(student.person_1, str);
                break;
    case 4 : strcpy(str, get_str(st_wh, NAME_LEN-1, lcol+10, 10, &last_key));
                if (strcmp(str, NULL))
                   strcpy(student.person_2, str);
                break;
    case 5 : strcpy(str, get_str(st_wh, DIAG_LEN-1, lcol+11, 12, &last_key));
             if (strcmp(str, NULL))
                strcpy(student.location, str);
             break;
       case 6 : wtext_out(st_wh, lcol, 14, "Exit and Save", reverse_color);
                last_key = 0;
          while (last_key != DOWN && last_key != RETURN && last_key != ESC && last_key != UP)
                   last_key = get_key();

                if (last_key == ESC || last_key == RETURN)
                   FINISHED = TRUE;
                break;
```

```
      case 7 : wtext_out(st_wh, lcol, 16, "Select Student", reverse_color);
               last_key = 0;
               while (last_key != RETURN && last_key != ESC && last_key != UP
                  last_key = get_key();

               if (last_key == ESC || last_key == RETURN)
                  FINISHED = TRUE;
               if (last_key == RETURN)
                   rc = 1;
               break;

     }/*switch*/

    switch(last_key)
     {
      case UP:      if (--item <0)
                      item = 0;
                    break;
      case RETURN:  if (++item > num_items)
                      item = num_items;
                    break;
      case DOWN:    if (++item > num_items)
                      item = num_items;
                    break;
      case ESC:     FINISHED = TRUE;
                    break;
     }/*switch*/

    }/*while*/


   WinDestroyWindow(st_wh);
   if (last_key == RETURN)
     save_student_info(OPERATION);

   free(str);
   if(rc)
       return student.name;
     else
       return "";


  }

  /**********************************************************************/
  /* FUNCTION   delete_file                                             */
  /* REMARKS    if the file or wildcard exists, then the file is        */
  /*            deleted without any screen output                       */
  /* INPUT      char *file - file to delete                             */
  /* OUTPUT     the success of the deletion                             */
  /* CALLED BY  delete_data_files                                       */
  /* CALLS                                                              */
  /**********************************************************************/

  int delete_file(char *file)
  {
```

```
 char del_cmd[] = "del ";
 char command[19];
 struct ffblk f;
 int a=0;

 if (findfirst(file, &f, a)==0)
  {
    strcpy(command, del_cmd);
    strcat(command, file);
    return system(command);
  }/*if*/
 else
  return -1;
}


/***********************************************************************/
/* FUNCTION   delete_data_files                                      */
/* REMARKS    deletes a file from a path; handles wildcards          */
/* INPUT      char *path - path to delete                            */
/*            int color - color to display the message               */
/* OUTPUT     int - success or not                                   */
/* CALLED BY  delete_student_named                                   */
/* CALLS                                                                  */
/***********************************************************************/

int delete_data_files(char *path, int color)
{
 int x=0, c;
 char str[100]="";
 extern char directory[50][14];

 strcpy(str, program_root_dir);
 strcat(str, "\\");
 strcat(str, path);

 chdir(str);
 c = get_filenames("c:", "*.dat");  /* the address is blown
                                      when this function returns */
 c-=2;
 textattr(color);
 while(x <= c)
  {
   strcpy(str, directory[x]);
   strcat(str, ".*");
   delete_file(str);
   x++;
  }/*while*/
 delete_file("*.dat");
 delete_file("*.vpc");
 delete_file("null.*");

 for (x=0;x<=9;x++)
  {
   sprintf(str, "*.%1d*", x);
   delete_file(str);
```

```
  }/*for*/
 delete_file("num.");
 chdir(program_root_dir);

 puts("Returning c");
 return c;
}

/**********************************************************************/
/* FUNCTION   delete_name_from_student_list                           */
/* REMARKS    deletes a student name from the studlist.dat file       */
/* INPUT      char *name - name of student to delete                  */
/* OUTPUT                                                             */
/* CALLED BY  delete_student_named                                    */
/* CALLS                                                              */
/**********************************************************************/

void delete_name_from_student_list(char *name)
{
 FILE *fp1, *fp2;
 char *sname;    /* student name read in from the file*/

 fp1 = fopen("studlist.dat", "rt");
 if (fp1)
  {
   sname = (char *) malloc (sizeof(char) * 9);
   strcpy(sname, "qwerty\n");
   fp2 = fopen("temp.dat", "wt");
   fgets(sname, 9, fp1);
   while(!feof(fp1) && strcmp(sname, NULL))
    {
     sname[strlen(sname)-1] = 0;
     if (strcmp(sname, name))
      {
       fputs(sname, fp2);
       fputc('\n', fp2);
      }/*if*/

     strcpy(sname, NULL);
     fgets(sname, 9, fp1);

    }/*while*/
   fflush(fp2);
   fclose(fp2);
   remove("studlist.dat");
   rename("temp.dat", "studlist.dat");
  }/*if*/

 fflush(fp1);
 fclose(fp1);
}

/**********************************************************************/
/* FUNCTION   delete_student_named                                    */
/* REMARKS    yes/no query to verify deletion of user                 */
/* INPUT      char *name - name of user to delete                     */
```

```
/* OUTPUT                                                              */
/* CALLED BY delete_or_edit_student                                    */
/* CALLS     delete_data_files, delete_name_from_student_list              */
/*********************************************************************/

void delete_student_named(char *name)
{
 HWND w_wh;
 char *buf;
 short DELETED = FALSE;

 buf = (char *) malloc (sizeof(char) * 50);
w_wh = WinCreateWindow(11, 10, 60, 5, warning_color, DBL_LINE_SIDES, warning_color, EXPLODE);
 sprintf(buf,  "Deleting %s will erase all of %s's files.", name, name);
 WinCenterText(w_wh, 1, buf, warning_color);
 WinSetCursorPos(w_wh, 3, 19);
 if (delete_data_files(name, warning_color) > -1)
    DELETED = TRUE;
   else
    DELETED = FALSE;

 printf("Returned from delete data files");
 get_key();

 WinDestroyWindow(w_wh);
 if (DELETED)
  {
   rmdir(name);
             /* delete name from studlist.dat file */
   delete_name_from_student_list(name);
  }/*if*/

 free(buf);
}

/*********************************************************************/
/* FUNCTION   delete_or_edit_student                                   */
/* REMARKS    command center for deleting or editing a student         */
/* INPUT      int OPERATION - DELETE or EDIT a student                 */
/* OUTPUT                                                                  */
/* CALLED BY  main                                                     */
/* CALLS      wtext_out, delete_student_named                          */
/*********************************************************************/
char* delete_or_edit_student(int OPERATION)
{
 FILE *fp;
 char *str;
 int c, x;
 HWND sl_wh;      /*student list window handle*/
 HWND w_wh;       /* warning window handle*/
 char st_name[NAME_LEN+1]="";

 str = (char *)malloc(sizeof(char) * NAME_LEN);

 fp = fopen("studlist.dat", "rt");
 if (fp)
```

```
{
 c = 0;
 while(!feof(fp) && c < MAX_STUDENTS)
  {
   fgets(name_list[c].str, NAME_LEN+1, fp);
   if (name_list[c].str)
   {
    name_list[c].str[strlen(name_list[c].str)-1]=0;
    c++;
   }/*for*/
  }/*while*/
 fflush(fp);
 fclose(fp);

 x = c - 1;
 c-=2;
 if (c>=0)
  {
sl_wh = WinCreateWindow(1, 1, 25, x+2, win_color, DBL_LINE_SIDES, bdr_color, EXPLODE);
    WinCenterText(sl_wh, 0, "Student List", win_color);
    while(c >= 0)
     {
       wtext_out(sl_wh, 2, c+1, name_list[c].str, win_color);
       name_list[c].col = 2;
       name_list[c].row = c+1;
       c--;
     }/*while*/
  }/*if*/

 if ( x > 0 )
 {
w_wh = WinCreateWindow(11, 10, 60, 3, message_color, DBL_LINE_SIDES, message_color, EXPLODE);
  switch (OPERATION)
    {
      case DELETE_STUDENT : WinCenterText(w_wh, 1,
               "Point to the student to delete.", message_color);
                           break;
      case EDIT_STUDENT   : WinCenterText(w_wh, 1,
               "Point to the student to edit.", message_color);
                           break;
    }/*switch*/

  if (get_key() != ESC)
   {
     WinDestroyWindow(w_wh);
     c = point_and_shoot(sl_wh, name_list, x);
     if (c>-1)
     switch(OPERATION)
       {
         case DELETE_STUDENT : delete_student_named(name_list[c-1].str);
                              break;
  case EDIT_STUDENT   : strcpy(st_name, add_or_edit_student(EDIT_STUDENT, name_list[c-1].str));
                              break;
       }/*switch*/
    }/*if*/
    else
```

BEST COPY AVAILABLE

```
        WinDestroyWindow(w_wh);

    }/*if*/
   else
    {
      put_message("No students found!");

    }/*else*/

   }/*if*/
   else
    put_message("No students found!");
  fclose(fp);
  free(str);
  WinDestroyWindow(sl_wh);
  if(strcmp(st_name, ""))
    return st_name;
   else
    return "";

  }

/***********************************************************************/
/* FUNCTION    put_message                                            */
/* REMARKS     puts a message in a window                             */
/* INPUT       char *str - message                                    */
/* OUTPUT                                                              */
/* CALLED BY  whether the user wishes to continue or ESC              */
/* CALLS      get_key                                                 */
/***********************************************************************/

int put_message(char *str)
{
HWND wh;
int key;

wh = WinCreateWindow(11, 10, 60, 3, message_color, DBL_LINE_SIDES, message_color, EXPLODE);
  WinCenterText(wh, 1, str, message_color);
  key = get_key();
  WinDestroyWindow(wh);
  if (key == ESC)
     return -1;
   else
     return 1;

}
/***********************************************************************/
/* FUNCTION    display_student_list                                   */
/* REMARKS     opens a window and displays a student list for         */
/*             point and shoot                                        */
/* INPUT                                                               */
/* OUTPUT      int - student id of the selected student               */
/* CALLED BY   select_student                                         */
/* CALLS       point_and_shoot, put_message                           */
/***********************************************************************/
```

```
int display_student_list(void)
{
 char *str;
 int c, x;
 FILE *fp;
 HWND sl_wh;

 str = (char *) malloc (sizeof(char) * 10);
 fp = fopen("studlist.dat", "rt");
 if (fp)
  {
   c = 0;
   while(!feof(fp) && c < MAX_STUDENTS)
    {
     fgets(name_list[c].str, NAME_LEN, fp);
     if (name_list[c].str)
     {
      name_list[c].str[strlen(name_list[c].str)-1]=0;
      c++;
     }/*for*/
    }/*while*/

   x = c - 1;
   c-=2;
   if (c>=0)
    {
  sl_wh = WinCreateWindow(1, 1, 25, x+2, win_color, DBL_LINE_SIDES, bdr_color, EXPLODE);
      WinCenterText(sl_wh, 0, "Student List", win_color);
      while(c >= 0)
       {
         wtext_out(sl_wh, 2, c+1, name_list[c].str, win_color);
         name_list[c].col = 2;
         name_list[c].row = c+1;
         c--;
       }/*while*/
    }/*if*/

   if ( x > 0  &&  put_message("Point to the desired student.") > -1)
       c = point_and_shoot(sl_wh, name_list, x);
     else
       if (x <= 0)
         put_message("No students found!");
 }/*if*/
 else
  {
    put_message( "No students found!");
    c = -1;
  }/*else*/

fclose(fp);
free(str);
WinDestroyWindow(sl_wh);
return c;
}
```

BEST COPY AVAILABLE

```
/**********************************************************************/
/* FUNCTION    select_student                                         */
/* REMARKS     prompts the user to select the student                 */
/* INPUT       char *name - student's name                            */
/* OUTPUT                                                                  */
/* CALLED BY   main                                                   */
/* CALLS       display_student_list                                   */
/**********************************************************************/

void select_student(char *name)
{
 int c;
 char buf[100];

 if (!strcmp(name, ""))
  {
   c = display_student_list();
   if (c> -1)
    {
     strcpy(buf, program_root_dir);
     strcat(buf, "\\");
     strcat(buf, name_list[c-1].str);
     chdir(buf);
     main_menu();
     textattr(reverse_color);
     clrscr();
    }/*if*/
   }/*if*/
   else
    {
     strcpy(buf, program_root_dir);
     strcat(buf, "\\");
     strcat(buf, name_list[c-1].str);
     chdir(buf);

     main_menu();
     textattr(reverse_color);
     clrscr();
    }/*else*/

}

void main(void)
{
 int  item = 0;
 int  c;
 char st_name[NAME_LEN+1]="";
 HWND wh;
 int  num_ts = 5;  /* num items on main menu*/
#define buflen 80
 char buf[buflen];

#if !DEBUG
 spawnlp(P_WAIT, "bird", 0);
#endif
```

```
   puts("Trying to initialize");
   WinInitialize();
   puts("Initialized");

   textattr(reverse_color);
   clrscr();
   title_screen();
   puts("Title Screen");
 name_list = (struct text_type*) malloc(sizeof(struct text_type) * MAX_STUDENTS);
   text1 = (struct text_type*) malloc(sizeof(struct text_type) * num_ts);
   student.name = (char *) malloc (sizeof(char) * NAME_LEN);
   student.age = (char *) malloc (sizeof(char) * AGE_LEN);
   student.diagnosis = (char *) malloc (sizeof(char) * DIAG_LEN);
   student.person_1 = (char *) malloc (sizeof(char) * NAME_LEN);
   student.person_2 = (char *) malloc (sizeof(char) * NAME_LEN);
   student.location = (char *) malloc(sizeof(char) * DIAG_LEN);

   for (c=0;c<MAX_STUDENTS;c++)
      name_list[c].str = (char*)malloc(sizeof(char) * NAME_LEN);
            /* allocate name list objects*/

   assign_menu_attribs();
   clear_student_space();

   while (item < num_ts )
   {
    textattr(reverse_color);
    clrscr();
    gotoxy(1,1);
    cprintf("%s   ", getcwd(buf, buflen));

    program_root_dir = (char *) malloc ( sizeof(char) * (strlen(buf)+1) );
    strcpy(program_root_dir, buf);

    open_and_print_student_window(&wh, num_ts);
    item = point_and_shoot(wh, text1, num_ts);
    WinDestroyWindow(wh);

    switch(item)
     {
      case -1: item = num_ts;
               break;
      case  1: strcpy(st_name, add_or_edit_student(ADD_STUDENT, ""));
               break;
      case  2: delete_or_edit_student(DELETE_STUDENT);
               break;
      case  3: strcpy(st_name, delete_or_edit_student(EDIT_STUDENT));
               break;
      case  4: select_student("");
               break;

     }/*switch*/

    if (strcmp(st_name, ""))
     {
       select_student(st_name);
```

```
      strcpy(st_name, "");
    }/*if*/

    free(program_root_dir);
  }/*while*/

  WinDestroyWindow(wh);
  WinTerminate();

  free_all(num_ts);
  flushall();
}
```

```
#include<process.h>
#include<string.h>
#include<stdio.h>
#include<alloc.h>
#include<conio.h>
#include<dir.h>
#include<win.h>
#include<pointsht.h>
#include<keyboard.h>
#include"student.h"

#define REPORT "Report"
#define STS  "STS"
#define SETUP  "Setup"
#define SPEECH_MGR "Speech Manager"
#define EXIT "Exit"
#define MAX_STUDENTS   20     /* max number of students*/
#define NAME_LEN 9
#define AGE_LEN 3
#define DIAG_LEN 21

extern short warning_color;
extern short win_color;
extern short bdr_color;
extern short EXPLODE;

struct text_type *text;
extern struct text_type *name_list;
struct student_type{
  char  *name;
  char  *age;
  char  *diagnosis;
  char  *person_1;
  char  *person_2;
 };

extern struct student_type student;
extern char * program_root_dir;

void wtext_out(HWND wh, int col, int row, char *str, int color);

/*******************************************************************/
/* FUNCTION   open_and_print_main_menu                            */
/* REMARKS    opens and prints the main window of the student      */
/*            program                                              */
/* INPUT      HWND *wh - pointer to the window handle              */
/*            int num_ts - number of menu items                    */
/* OUTPUT                                                              */
/* CALLED BY  main                                                 */
/* CALLS      wtext_out                                         */
/*******************************************************************/

void open_and_print_main_menu(HWND *wh, int num_ts)
{
int c;
```

```
*wh = WinCreateWindow(7, 15, 50, 12, win_color, DBL_LINE_ALL_SIDES, bdr_color, EXPLODE);
  WinCenterText(*wh, 0, "Sound to Speech", win_color);
  for (c=0;c<num_ts;c++)
    wtext_out(*wh, text[c].col, text[c].row, text[c].str, win_color);

}

/***********************************************************************/
/* FUNCTION   assign_central_menu_attribs                           */
/* REMARKS    assigns the menu attributes for the main menu to      */
/*            allow for point and shoot routine                     */
/* INPUT                                                                 */
/* OUTPUT                                                                */
/* CALLED BY  main                                                  */
/* CALLS      allocate_text_object                                  */
/***********************************************************************/

void assign_central_menu_attribs(void)
{
int lcol = 2;
int first_row=2;

 allocate_text_object(text, 0, STS, lcol, first_row);
 allocate_text_object(text, 1, SETUP, lcol, first_row+2);
 allocate_text_object(text, 2, SPEECH_MGR, lcol, first_row+4);
 allocate_text_object(text, 3, REPORT, lcol, first_row+6);
 allocate_text_object(text, 4, EXIT, lcol, first_row+8);

}

/***********************************************************************/
/* FUNCTION   speech_mgr                                            */
/* REMARKS    prompts the user to choose between Recognition        */
/*            functions and Recording Functions.                    */
/* INPUT                                                                 */
/* OUTPUT                                                                */
/* CALLED BY  main_menu                                             */
/* CALLS      allocate_text_object                                  */
/***********************************************************************/


void speech_mgr(void)
{
int k=0;
struct text_type *text;
HWND wh;
int c=0;

#define RECOGNITION  "Recognition Functions"
#define RECORDING    "Recording Functions"


 text = (struct text_type*) malloc(sizeof(struct text_type) * 3);
 allocate_text_object(text, 0, RECOGNITION, 2, 2);
 allocate_text_object(text, 1, RECORDING, 2, 4);
 allocate_text_object(text, 2, EXIT, 2, 7);
```

```
 wh = WinCreateWindow(10, 27, 26, 9, win_color, DBL_LINE_ALL_SIDES, bdr_color, EXPLODE);
  WinCenterText(wh, 0, "Speech Manager", win_color);
  for (c=0;c<=2;c++)
    wtext_out(wh, text[c].col, text[c].row, text[c].str, win_color);

  k = point_and_shoot(wh, text, 3);
  WinDestroyWindow(wh);

  for (c=0;c<=2;c++)
    free(text[c].str);
  free(text);
  switch(k)
   {
      case 1: spawnlp(P_WAIT, "strain", 0);
                break;
      case 2: spawnlp(P_WAIT, "speech", 0);
                break;

   }/*switch*/

 }

/**********************************************************************/
/* FUNCTION   main_menu                                               */
/* REMARKS    main menu for student program.  From here you can       */
/*            run STSMAIN, SETUP, SPEECH MANAGER, or return to the    */
/*            main system menu                                        */
/* INPUT                                                                   */
/* OUTPUT                                                                  */
/* CALLED BY main                                                     */
/* CALLS      assign_central_menu_attribs, open_and_print_main_menu */
/*            speech_mgr                                              */
/**********************************************************************/


void main_menu(void)
{
 char buf[80];
 int buflen=80;
 HWND wh;
 int  item=0, num_ts = 5;
 extern short reverse_color;

 text = (struct text_type*) malloc(sizeof(struct text_type) * 5);

 assign_central_menu_attribs();
 while (item < num_ts && item != -1 )
  {
   textattr(reverse_color);
   clrscr();

   gotoxy(1,1);
   cprintf("%s   ", getcwd(buf, buflen));

   open_and_print_main_menu(&wh, num_ts);
```

```
    item = point_and_shoot(wh, text, num_ts);
    WinDestroyWindow(wh);

    switch(item)
     {
      case -1: item = num_ts;
               break;
      case  1: spawnlp(P_WAIT, "stsmain", 0);
               break;
      case  2: spawnlp(P_WAIT, "setup", 0);
               break;
      case  3: speech_mgr();
                break;
      case  4: spawnlp(P_WAIT, "report", 0);
               break;
     }/*switch*/

   }/*while*/

  chdir(program_root_dir);
  for(item=0;item<=4;item++)
    free(text[item].str);
  free(text);
  }


  /*********************************************************************/
  /* FUNCTION    title_screen()                                        */
  /* REMARKS     explodes the title screen                             */
  /* INPUT                                                             */
  /* OUTPUT                                                            */
  /* CALLED BY   main                                                  */
  /* CALLS       get_key()                                             */
  /*********************************************************************/

  void title_screen(void)
  {
   HWND wh;
   char ver[5]="", buf[14]="Version ";
   FILE *fp;

 wh = WinCreateWindow(5, 10, 60, 15, win_color, DBL_LINE_ALL_SIDES, bdr_color, EXPLODE);
   WinCenterText(wh, 1, "Sound To Speech", win_color);
   fp = fopen("version.dat", "rt");
   if (fp)
      fgets(ver, 6, fp);
   fclose(fp);
   strcat(buf, ver);

   WinCenterText(wh, 3, buf, win_color);
   WinCenterText(wh, 5, "Copyright (C) ARC of the US, 1988-1991", win_color);
 WinCenterText(wh, 8, "Copyright (C) Genus Microprogramming, Inc. 1988-1990", win_color);
 WinCenterText(wh, 10, "Copyright (C) Greenleaf Software, Inc. 1984-1989", win_color);
   WinCenterText(wh, 13, "Press a key to continue", win_color);
```

```
    get_key();
    WinDestroyWindow(wh);

}
```

```
main()
{
  printf("\n\n%d\n\n", sizeof(char));
}
```

```
#define DEBUG 1

#include<conio.h>
#include<string.h>
#include<alloc.h>
#include<process.h>
#include<dir.h>
#include<stdio.h>
#include"win.h"
#include"pointsht.h"
#include"keyboard.h"
#include"student.h"
#define BLANK_STR_LEN_8 "        "

char *program_root_dir; /* sts package root directory */
char name_buf[30];
void clear_student_space(void);
int  put_message(char *);
void main_menu(void);
int  get_filenames(char *, char *);
void title_screen(void);

#define ADD_STUDENT    2
#define SELECT_STUDENT 3
#define DELETE_STUDENT 0
#define EDIT_STUDENT   1
#define MAX_STUDENTS   20     /* max number of students*/
#define NAME_LEN 21
#define AGE_LEN 3
#define DIAG_LEN 24
#define ADD_ST  "Add Student"
#define SELECT_ST  "Select Student"
#define DEL_ST  "Delete Student"
#define EDIT_ST "Edit Student Description"
#define EXIT_ST "Exit"

short message_color = WHITE|REV_BLUE|HI_INTENSITY;
short warning_color = WHITE|REV_RED|HI_INTENSITY;
short win_color = WHITE|REV_GREEN|HI_INTENSITY;
short bdr_color = WHITE|REV_GREEN|HI_INTENSITY;
short reverse_color = WHITE|HI_INTENSITY;
short EXPLODE = TRUE;

FILE *fopen();

struct text_type *text1;
struct text_type *name_list;

struct student_type{
  char  *name;
  char  *age;
  char  *diagnosis;
  char  *person_1;
  char  *person_2;
  char  *location;
 };
```

```
struct student_type student;

/*****************************************************************/
/* FUNCTION   free_all                                           */
/* REMARKS    frees all allocated memory space allocated by malloc */
/* INPUT      int num_ts - number of main menu items             */
/* OUTPUT                                                        */
/* CALLED BY  main                                               */
/* CALLS                                                         */
/*****************************************************************/

void free_all(int num_ts)
{
int c;

 flushall();
 fcloseall();
 for (c=0;c<num_ts;c++)
  free(text1[c].str);

 for (c=0;c<MAX_STUDENTS;c++)
   free(name_list[c].str);
 free(student.name);
 free(student.age);
 free(student.diagnosis);
 free(student.person_1);
 free(student.person_2);
 free(student.location);
 free(text1);
 free(name_list);
}

/*****************************************************************/
/* FUNCTION   wtext_out                                          */
/* REMARKS    Prints text to a window at a specified row, col and */
/*            color                                              */
/* INPUT      HWND wh - window handle to print to                */
/*            int col - column of window to print at             */
/*            int row - row of window to print at                */
/*            char *str - string to print                        */
/*            int color - color attributes for the string to print */
/* OUTPUT                                                        */
/* CALLED BY  open_and_print_student_window, display_st_info_str, */
/*            create_add_student_window, add_or_edit_student,    */
/*            delete_or_edit_student                             */
/* CALLS                                                         */
/*****************************************************************/

void wtext_out(HWND wh, int col, int row, char *str, int color)
{
 WinSetCursorPos(wh, row, col);
 WinTextOut(wh, str, color);
}

/*****************************************************************/
```

```
/* FUNCTION   put_warning_msg                                          */
/* REMARKS    Opens a red and white window at the center of the       */
/*            screen, and prints a message                            */
/* INPUT      char *str - string or message to print                  */
/* OUTPUT                                                             */
/* CALLED BY  save_student_info                                       */
/* CALLS      get_key                                                 */
/*********************************************************************/

void put_warning_msg(char *str)
{
 HWND wh;

wh = WinCreateWindow(11, 10, 60, 3, warning_color, DBL_LINE_SIDES, warning_color, EXPLODE);
 WinCenterText(wh, 1, str, warning_color);
 get_key();
 WinDestroyWindow(wh);
}

/*********************************************************************/
/* FUNCTION   assign_menu_attribs                                     */
/* REMARKS    assigns the menu attributes for the main menu to        */
/*            allow for point and shoot routine                       */
/* INPUT                                                              */
/* OUTPUT                                                             */
/* CALLED BY  main                                                    */
/* CALLS      allocate_text_object                                    */
/*********************************************************************/


void assign_menu_attribs(void)
{
int first_row = 2;
int lcol = 2;

 allocate_text_object(text1, 0, ADD_ST, lcol, first_row);
 allocate_text_object(text1, 1, DEL_ST, lcol, first_row+2);
 allocate_text_object(text1, 2, EDIT_ST, lcol, first_row+4);
 allocate_text_object(text1, 3, SELECT_ST, lcol, first_row+6);
 allocate_text_object(text1, 4, EXIT_ST, lcol, first_row+10);

}


/*********************************************************************/
/* FUNCTION   open_and_print_student_window                           */
/* REMARKS    opens and prints the main window of the student         */
/*            program                                                 */
/* INPUT      HWND *wh - pointer to the window handle                 */
/*            int num_ts - number of menu items                       */
/* OUTPUT                                                             */
/* CALLED BY  main                                                    */
/* CALLS      wtext_out                                               */
/*********************************************************************/

void open_and_print_student_window(HWND *wh, int num_ts)
```

```
{
int c;

*wh = WinCreateWindow(7, 15, 50, 14, win_color, DBL_LINE_ALL_SIDES, bdr_color, EXPLODE);
 WinCenterText(*wh, 0, "Student", win_color);
 for (c=0;c<num_ts;c++)
  wtext_out(*wh, text1[c].col, text1[c].row, text1[c].str, win_color);

}


/**********************************************************************/
/* FUNCTION    create_add_student_window                              */
/* REMARKS     creates and prints the text1 for the attribute input   */
/* INPUT       HWND *st_wh - pointer to the student window            */
/*             int lcol - left column the attribute names will be     */
/*             printed at                                             */
/* OUTPUT                                                             */
/* CALLED BY   add_or_edit_student                                    */
/* CALLS       wtext_out                                              */
/**********************************************************************/

void create_add_student_window(int OPERATION, HWND *st_wh, int lcol)
{

*st_wh = WinCreateWindow(7, 15, 50, 18, win_color, DBL_LINE_SIDES, bdr_color, EXPLODE);
 switch(OPERATION)
  {
   case ADD_STUDENT : WinCenterText(*st_wh, 0, "Add Student", win_color);
                      break;
   case EDIT_STUDENT: WinCenterText(*st_wh, 0, "Edit Student", win_color);
                      break;
  }/*switch*/

 wtext_out(*st_wh, lcol, 2, "Name:[", win_color);
 wtext_out(*st_wh, lcol+5+NAME_LEN, 2, "]", win_color);
 wtext_out(*st_wh, lcol, 4, "Age:[", win_color);
 wtext_out(*st_wh, lcol+4+AGE_LEN, 4, "]", win_color);
 wtext_out(*st_wh, lcol, 6, "Diagnosis:[", win_color);
 wtext_out(*st_wh, lcol+10+DIAG_LEN, 6, "]", win_color);
 wtext_out(*st_wh, lcol, 8, "Person 1:[", win_color);
 wtext_out(*st_wh, lcol+9+NAME_LEN, 8, "]", win_color);
 wtext_out(*st_wh, lcol, 10, "Person 2:[", win_color);
 wtext_out(*st_wh, lcol+9+NAME_LEN, 10, "]", win_color);
 wtext_out(*st_wh, lcol, 12, "Location :[", win_color);
 wtext_out(*st_wh, lcol+DIAG_LEN+10, 12,  "]", win_color);
 wtext_out(*st_wh, lcol, 14, "Exit and Save", win_color);
 wtext_out(*st_wh, lcol, 16, "Select Student", win_color);


 /*---------------------clear out new space ---------------*/
 clear_student_space();

}
/**********************************************************************/
/* FUNCTION   display_st_info_str                                     */
```

```
/* REMARKS      displays the individual attributes of each student   */
/* INPUT        HWND wh - window handle to print to                  */
/*              char *str - character string to print                */
/*              char *blank - blank string                           */
/*              int col - column to print at                         */
/*              int row - row to print at                            */
/* OUTPUT                                                            */
/* CALLED BY    display_student_info                                 */
/* CALLS        wtext_out                                            */
/*********************************************************************/


void display_st_info_str(HWND wh, char *str, char *blank, int col, int row)
{
 if (strcmp(str, NULL) && strcmp(str, "-"))
   wtext_out(wh, col, row, str, win_color);
  else
   wtext_out(wh, col, row, blank,  win_color);

}

/*********************************************************************/
/* FUNCTION     display_student_info                                 */
/* REMARKS      center for printing student attributes               */
/* INPUT        HWND wh - window to print to                         */
/*              int lcol - column to print at                        */
/* OUTPUT                                                            */
/* CALLED BY    add_or_edit_student                                  */
/* CALLS        display_st_info_str                                  */
/*********************************************************************/


void display_student_info(HWND wh, int lcol)
{
 display_st_info_str(wh, student.name, BLANK_STR_LEN_8, lcol+6, 2);
 display_st_info_str(wh, student.age, "  ", lcol+5, 4);
 display_st_info_str(wh, student.diagnosis, BLANK_STR_LEN_8, lcol+11, 6);
 display_st_info_str(wh, student.person_1, BLANK_STR_LEN_8, lcol+10, 8);
 display_st_info_str(wh, student.person_2, BLANK_STR_LEN_8, lcol+10, 10);
 display_st_info_str(wh, student.location, "           ", lcol+11, 12);


 wtext_out(wh, lcol, 14, "Exit and Save", win_color);
 wtext_out(wh, lcol, 16, "Select Student", win_color);


}
/*********************************************************************/
/* FUNCTION     write_str2file                                       */
/* REMARKS      If the string exists then the string is written to   */
/*              a file, else a "-" is written.                       */
/* INPUT        FILE *fp - file to write to                          */
/*              char *str - string to write                          */
/* OUTPUT                                                            */
/* CALLED BY    update_student_list_file, save_student_info          */
/* CALLS                                                             */
```

```
/*********************************************************************/

void write_str2file(FILE *fp, char *str)
{
if (str && strlen(str)>0)
  {
   fprintf(fp, "%s\n", str);
  }/*if*/
 else
  {
   fprintf(fp, "-\n");
  }/*else*/

}
/*********************************************************************/
/* FUNCTION    update_student_list_file                              */
/* REMARKS     updates the student list by adding the new name       */
/* INPUT       FILE *fp - file handle for the student list file      */
/* OUTPUT                                                            */
/* CALLED BY   save_student_info                                     */
/* CALLS       write_str2file,                                       */
/*********************************************************************/


void update_student_list_file(void)
{
FILE *fp1;
char str[10]="";
int FOUND = FALSE;

  /* check to ensure that the name does not exist in the file */
  fp1 = fopen("studlist.dat", "rt");
  if (fp1)
   {
     while(!feof(fp1) && strcmp(str, ""))
     {
       fgets(str, 10, fp1);
       str[strlen(str)-1]=0;
       if(!strcmp(str, student.name))
           FOUND = TRUE;
     }/*while*/
   }/*if*/
  fflush(fp1);
  fclose(fp1);


  if (!FOUND)
   {
     fp1 = fopen("studlist.dat", "at");
     if (fp1)
      {
       write_str2file(fp1, student.name);

      }/*if*/
     fflush(fp1);
     fclose(fp1);
```

```
  }/*if*/
}
/*********************************************************************/
/* FUNCTION   save_student_info                                      */
/* REMARKS    saves the student attributes to a file                 */
/* INPUT                                                             */
/* OUTPUT                                                            */
/* CALLED BY                                               */
/* CALLS      put_warinng_msg, write_str2file                        */
/*********************************************************************/

void save_student_info(int OPERATION)
{
  FILE *fp;
  char buf[100];
  char buf1[75];
  char dir_name[9];
  char ins_dash[2];
  char *blnk_pt;
  char fst_prt[5], lst_prt[5];

  strcpy(buf, program_root_dir);
  strncpy(dir_name, student.name,8);
  blnk_pt = strstr(dir_name," ");
  ins_dash[0] ='-';
  if (blnk_pt != NULL)
  {
    strncpy(fst_prt,strtok(dir_name," "),4);
    strncpy(lst_prt,strtok(NULL," "),4);
    strncpy(dir_name,fst_prt,4);
    strncat(dir_name,ins_dash,1);
    strncat(dir_name,lst_prt,4);
  }
  if (mkdir(dir_name) && OPERATION == ADD_STUDENT)
    put_warning_msg("You need a unique name!");
  else
   {
    strcat(buf, "\\");
    strcat(buf, dir_name);
    chdir(buf);
    fp = fopen("students.dat", "wt");
    if (fp)
    {
     write_str2file(fp, student.name);
     write_str2file(fp, student.age);
     write_str2file(fp, student.diagnosis);
     write_str2file(fp, student.person_1);
     write_str2file(fp, student.person_2);
     write_str2file(fp, student.location);
    }/*if*/
    fflush(fp);
    fclose(fp);
    chdir(program_root_dir);
    if (OPERATION == ADD_STUDENT)
      update_student_list_file();
   }/*else*/
```

```
}

/*********************************************************************/
/* FUNCTION   clear_student_space                                    */
/* REMARKS    sets all student attributes to null                    */
/* INPUT                                                             */
/* OUTPUT                                                            */
/* CALLED BY  create_add_student_window, add_or_edit_student         */
/* CALLS                                                             */
/*********************************************************************/

void clear_student_space(void)
{
 strcpy(student.name, "-");
 strcpy(student.age, "-");
 strcpy(student.diagnosis, "-");
 strcpy(student.person_1, "-");
 strcpy(student.person_2, "-");
 strcpy(student.location, "-");
}

/*********************************************************************/
/* FUNCTION   read_str_from_file                                     */
/* REMARKS    reads a string from a file                             */
/* INPUT      FILE *fp - file to read from                           */
/*            char *str - pointer to string to return                */
/* OUTPUT                                                            */
/* CALLED BY  read_student_data                                      */
/* CALLS                                                             */
/*********************************************************************/

void read_str_from_file(FILE *fp, char *str)
{
 fgets(str, NAME_LEN+1, fp);
 str[strlen(str)-1] = 0;

}

/*********************************************************************/
/* FUNCTION    check4null                                            */
/* REMARKS     checks to see if the string is blank                  */
/* INPUT       char *str - string to check                           */
/* OUTPUT                                                            */
/* CALLED BY   read_student_data                                     */
/* CALLS                                                             */
/*********************************************************************/

void check4null(char *str)
{
 if (!strcmp(str, "-"))
  strcpy(str, NULL);
}

/*********************************************************************/
/* FUNCTION   read_student_data                                      */
```

```
/* REMARKS      reads in the student data from the file              */
/* INPUT        char *name - the students name/directory to read from*/
/* OUTPUT                                                            */
/* CALLED BY    add_or_edit_student                                  */
/* CALLS        read_str_from_file, read_student_data                */
/********************************************************************/

void read_student_data(char *name)
{
 FILE *fp;
 char str[NAME_LEN]="", buf[100];
 name = &name_buf[0];
 strcpy(buf, program_root_dir);
 strcat(buf, "\\");
 strcat(buf, name);

 chdir(buf);
 fp = fopen("students.dat", "rt");

 read_str_from_file(fp, str);
 strcpy(student.name, str);
 check4null(student.name);

 read_str_from_file(fp, str);
 strcpy(student.age, str);
 check4null(student.age);

 read_str_from_file(fp, str);
 strcpy(student.diagnosis, str);
 check4null(student.diagnosis);

 read_str_from_file(fp, str);
 strcpy(student.person_1, str);
 check4null(student.person_1);

 read_str_from_file(fp, str);
 strcpy(student.person_2, str);
 check4null(student.person_2);

 read_str_from_file(fp, str);
 strcpy(student.location, str);
 check4null(student.location);

 fflush(fp);
 fclose(fp);
 chdir(program_root_dir);
}

/********************************************************************/
/* FUNCTION     add_or_edit_student                                  */
/* REMARKS      control center for adding a student to the STS system*/
/* INPUT        int OPERATION - ADD or EDIT                          */
/*              char *name - name of student to edit                 */
/* OUTPUT                                                            */
/* CALLED BY    main                                                 */
/* CALLS        create_add_student_window, clear_student_space,      */
```

```
/*              display_student_info, wtext_out, read_student_data   */
/************************************************************************/

char* add_or_edit_student(int OPERATION, char *name)
{
 HWND st_wh;
 int  lcol = 2, num_items = 7;
 char *str;
 int  item = 0;
 int  FINISHED = FALSE;
 int  last_key = 0;
 int rc = 0;

 str = (char *)malloc(sizeof(char) * NAME_LEN);

 create_add_student_window(OPERATION, &st_wh, lcol);
 switch(OPERATION)
  {
    case ADD_STUDENT:  clear_student_space();
                       break;

    case EDIT_STUDENT: read_student_data(name);
                       break;

  }/*switch*/
 textattr(win_color);
 while(!FINISHED)
  {
  display_student_info(st_wh, lcol);
  switch(item)
   {
     case 0 : if (OPERATION != EDIT_STUDENT)
                       {
                               strcpy(str, get_str(st_wh, NAME_LEN-1, lcol+6,
, &last_key));
                               if (strcmp(str, NULL))
                                       strcpy(student.name, str);
                              break;
                       }/*else*/
                       else
                               item=1;

    case 1 : strcpy(str, get_str(st_wh, AGE_LEN-1, lcol+5, 4, &last_key));
          if (strcmp(str, NULL))
             strcpy(student.age, str);
          break;
    case 2 : strcpy(str, get_str(st_wh, DIAG_LEN-1, lcol+11, 6, &last_key));
          if (strcmp(str, NULL))
             strcpy(student.diagnosis, str);
          break;
    case 3 : strcpy(str, get_str(st_wh, NAME_LEN-1, lcol+10, 8, &last_key));
          if (strcmp(str, NULL))
             strcpy(student.person_1, str);
          break;
 case 4 : strcpy(str, get_str(st_wh, NAME_LEN-1, lcol+10, 10, &last_key));
          if (strcmp(str, NULL))
             strcpy(student.person_2, str);
```

```
            break;
     case 5 : strcpy(str, get_str(st_wh, DIAG_LEN-1, lcol+11, 12, &last_key));
            if (strcmp(str, NULL))
               strcpy(student.location, str);
            break;
      case 6 : wtext_out(st_wh, lcol, 14, "Exit and Save", reverse_color);
            last_key = 0;
      while (last_key != DOWN && last_key != RETURN && last_key != ESC && last_key != UP)
               last_key = get_key();

            if (last_key == ESC || last_key == RETURN)
               FINISHED = TRUE;
            break;
      case 7 : wtext_out(st_wh, lcol, 16, "Select Student", reverse_color);
            last_key = 0;
            while (last_key != RETURN && last_key != ESC && last_key != UP)
               last_key = get_key();

            if (last_key == ESC || last_key == RETURN)
               FINISHED = TRUE;
            if (last_key == RETURN)
                rc = 1;
            break;

     }/*switch*/

    switch(last_key)
     {
      case UP:     if (--item <0)
                  item = 0;
                break;
      case RETURN: if (++item > num_items)
                  item = num_items;
                break;
      case DOWN:   if (++item > num_items)
                  item = num_items;
                break;
      case ESC:    FINISHED = TRUE;
                break;
     }/*switch*/

    }/*while*/

   WinDestroyWindow(st_wh);
   if (last_key == RETURN)
     save_student_info(OPERATION);

   free(str);
   if(rc)
       return student.name;
     else
       return "";

}
```

BEST COPY AVAILABLE

```
/**********************************************************************/
/* FUNCTION    delete_file                                            */
/* REMARKS     if the file or wildcard exists, then the file is       */
/*             deleted without any screen output                      */
/* INPUT       char *file - file to delete                            */
/* OUTPUT      the success of the deletion                            */
/* CALLED BY   delete_data_files                                      */
/* CALLS                                                              */
/**********************************************************************/

int delete_file(char *file)
{
 char del_cmd[] = "del ";
 char command[19];
 struct ffblk f;
 int a=0;

 if (findfirst(file, &f, a)==0)
  {
    strcpy(command, del_cmd);
    strcat(command, file);
    return system(command);
  }/*if*/
 else
  return -1;
}


/**********************************************************************/
/* FUNCTION    delete_data_files                                      */
/* REMARKS     deletes a file from a path; handles wildcards          */
/* INPUT       char *path - path to delete                            */
/*             int color - color to display the message               */
/* OUTPUT      int - success or not                                   */
/* CALLED BY   delete_student_named                                   */
/* CALLS                                                              */
/**********************************************************************/

int delete_data_files(char *path, int color)
{
 int x=0, c;
 char str[100]="";
 extern char directory[50][14];

 strcpy(str, program_root_dir);
 strcat(str, "\\");
 strcat(str, path);

 chdir(str);
 c = get_filenames("c:", "*.dat");  /* the address is blown
                                       when this function returns */
 c-=2;
 textattr(color);
 while(x <= c)
  {
   strcpy(str, directory[x]);
```

```
    strcat(str, ".*");
    delete_file(str);
    x++;
   }/*while*/
  delete_file("*.dat");
  delete_file("*.vpc");
  delete_file("null.*");

  for (x=0;x<=9;x++)
   {
    sprintf(str, "*.%1d*", x);
    delete_file(str);
   }/*for*/
  delete_file("num.");
  chdir(program_root_dir);

  puts("Returning c");
  return c;
 }

/*********************************************************************/
/* FUNCTION   delete_name_from_student_list                          */
/* REMARKS    deletes a student name from the studlist.dat file      */
/* INPUT      char *name - name of student to delete                 */
/* OUTPUT                                                            */
/* CALLED BY  delete_student_named                                   */
/* CALLS                                                             */
/*********************************************************************/

void delete_name_from_student_list(char *name)
{
 FILE *fp1, *fp2;
 char *sname;   /* student name read in from the file*/

 fp1 = fopen("studlist.dat", "rt");
 if (fp1)
  {
   sname = (char *) malloc (sizeof(char) * 9);
   strcpy(sname, "qwerty\n");
   fp2 = fopen("temp.dat", "wt");
   fgets(sname, 9, fp1);
   while(!feof(fp1) && strcmp(sname, NULL))
    {
     sname[strlen(sname)-1] = 0;
     if (strcmp(sname, name))
      {
       fputs(sname, fp2);
       fputc('\n', fp2);
      }/*if*/

     strcpy(sname, NULL);
     fgets(sname, 9, fp1);

    }/*while*/
   fflush(fp2);
   fclose(fp2);
```

```
    remove("studlist.dat");
    rename("temp.dat", "studlist.dat");
   }/*if*/

  fflush(fp1);
  fclose(fp1);
 }

/*********************************************************************/
/* FUNCTION  delete_student_named                                    */
/* REMARKS   yes/no query to verify deletion of user                 */
/* INPUT     char *name - name of user to delete                     */
/* OUTPUT                                                            */
/* CALLED BY delete_or_edit_student                                  */
/* CALLS     delete_data_files, delete_name_from_student_list              */
/*********************************************************************/

void delete_student_named(char *name)
{
 HWND w_wh;
 char *buf;
 short DELETED = FALSE;

 buf = (char *) malloc (sizeof(char) * 50);
w_wh = WinCreateWindow(11, 10, 60, 5, warning_color, DBL_LINE_SIDES, warning_color, EXPLODE);
 sprintf(buf,  "Deleting %s will erase all of %s's files.", name, name);
 WinCenterText(w_wh, 1, buf, warning_color);
 WinSetCursorPos(w_wh, 3, 19);
 if (delete_data_files(name, warning_color) > -1)
    DELETED = TRUE;
   else
    DELETED = FALSE;

 printf("Returned from delete data files");
 get_key();

 WinDestroyWindow(w_wh);
 if (DELETED)
  {
   rmdir(name);
          /* delete name from studlist.dat file */
   delete_name_from_student_list(name);
  }/*if*/

 free(buf);
}

/*********************************************************************/
/* FUNCTION   delete_or_edit_student                                 */
/* REMARKS    command center for deleting or editing a student       */
/* INPUT      int OPERATION - DELETE or EDIT a student               */
/* OUTPUT                                                            */
/* CALLED BY  main                                                   */
/* CALLS      wtext_out, delete_student_named                        */
/*********************************************************************/
char* delete_or_edit_student(int OPERATION)
```

```
{
 FILE *fp;
 char *str;
 int c, x;
 HWND sl_wh;      /*student list window handle*/
 HWND w_wh;       /* warning window handle*/
 char st_name[NAME_LEN+1]="";

 str = (char *)malloc(sizeof(char) * NAME_LEN);

 fp = fopen("studlist.dat", "rt");
 if (fp)
  {
   c = 0;
   while(!feof(fp) && c < MAX_STUDENTS)
    {
     fgets(name_list[c].str, NAME_LEN+1, fp);
     if (name_list[c].str)
     {
      name_list[c].str[strlen(name_list[c].str)-1]=0;
      c++;
     }/*for*/
    }/*while*/
   fflush(fp);
   fclose(fp);

   x = c - 1;
   c-=2;
   if (c>=0)
    {
  sl_wh = WinCreateWindow(1, 1, 25, x+2, win_color, DBL_LINE_SIDES, bdr_color, EXPLODE);
      WinCenterText(sl_wh, 0, "Student List", win_color);
      while(c >= 0)
        {
      wtext_out(sl_wh, 2, c+1, name_list[c].str, win_color);
      name_list[c].col = 2;
      name_list[c].row = c+1;
      c--;
        }/*while*/
    }/*if*/

   if ( x > 0 )
   {
 w_wh = WinCreateWindow(11, 10, 60, 3, message_color, DBL_LINE_SIDES, message_color, EXPLODE);
    switch (OPERATION)
      {
     case DELETE_STUDENT : WinCenterText(w_wh, 1,
                           "Point to the student to delete.", message_color)
                           break;
     case EDIT_STUDENT   : WinCenterText(w_wh, 1,
                           "Point to the student to edit.", message_color);
                           break;
      }/*switch*/

    if (get_key() != ESC)
     {
```

```
        WinDestroyWindow(w_wh);
        c = point_and_shoot(sl_wh, name_list, x);
        if (c>-1)
        switch(OPERATION)
       {
         case DELETE_STUDENT : delete_student_named(name_list[c-1].str);
                               break;
    case EDIT_STUDENT   : strcpy(st_name, add_or_edit_student(EDIT_STUDENT, name_list[c-1].str));
                               break;
       }/*switch*/
       }/*if*/
       else
        WinDestroyWindow(w_wh);

   }/*if*/
  else
   {
     put_message("No students found!");

   }/*else*/

  }/*if*/
  else
   put_message("No students found!");
 fclose(fp);
 free(str);
 WinDestroyWindow(sl_wh);
 if(strcmp(st_name, ""))
   return st_name;
  else
   return "";

 }

/*********************************************************************/
/* FUNCTION    put_message                                          */
/* REMARKS     puts a message in a window                           */
/* INPUT       char *str - message                                  */
/* OUTPUT                                                           */
/* CALLED BY  whether the user wishes to continue or ESC            */
/* CALLS      get_key                                               */
/*********************************************************************/

int put_message(char *str)
{
HWND wh;
int key;

wh = WinCreateWindow(11, 10, 60, 3, message_color, DBL_LINE_SIDES, message_color, EXPLODE);
  WinCenterText(wh, 1, str, message_color);
  key = get_key();
  WinDestroyWindow(wh);
  if (key == ESC)
     return -1;
   else
     return 1;
```

```
}
/*********************************************************************/
/* FUNCTION    display_student_list                                  */
/* REMARKS     opens a window and displays a student list for        */
/*             point and shoot                                       */
/* INPUT                                                             */
/* OUTPUT      int - student id of the selected student              */
/* CALLED BY   select_student                                        */
/* CALLS       point_and_shoot, put_message                          */
/*********************************************************************/

int display_student_list(void)
{
 char *str;
 int c, x;
 FILE *fp;
 HWND sl_wh;

 str = (char *) malloc (sizeof(char) * 10);
 fp = fopen("studlist.dat", "rt");
 if (fp)
  {
   c = 0;
   while(!feof(fp) && c < MAX_STUDENTS)
    {
     fgets(name_list[c].str, NAME_LEN, fp);
     if (name_list[c].str)
     {
      name_list[c].str[strlen(name_list[c].str)-1]=0;
      c++;
     }/*for*/
    }/*while*/

   x = c - 1;
   c-=2;
   if (c>=0)
    {
  sl_wh = WinCreateWindow(1, 1, 25, x+2, win_color, DBL_LINE_SIDES, bdr_color, EXPLODE);
      WinCenterText(sl_wh, 0, "Student List", win_color);
      while(c >= 0)
       {
      wtext_out(sl_wh, 2, c+1, name_list[c].str, win_color);
      name_list[c].col = 2;
      name_list[c].row = c+1;
      c--;
       }/*while*/
    }/*if*/

   if ( x > 0  &&  put_message("Point to the desired student.") > -1)
       c = point_and_shoot(sl_wh, name_list, x);
     else
       if (x <= 0)
      put_message("No students found!");
 }/*if*/
 else
```

```
    {
     put_message( "No students found!");
     c = -1;
    }/*else*/

  fclose(fp);
  free(str);
  WinDestroyWindow(sl_wh);
  return c;
  }


/***************************************************************************/
/* FUNCTION    select_student                                             */
/* REMARKS     prompts the user to select the student                     */
/* INPUT       char *name - student's name                                */
/* OUTPUT                                                                 */
/* CALLED BY   main                                                       */
/* CALLS       display_student_list                                       */
/***************************************************************************/

void select_student(char *name)
{
 int c;
 char buf[100];

 if (!strcmp(name, ""))
  {
   c = display_student_list();
   if (c> -1)
    {
     strcpy(buf, program_root_dir);
     strcat(buf, "\\");
     strcat(buf, name_list[c-1].str);
     chdir(buf);
     main_menu();
     textattr(reverse_color);
     clrscr();
    }/*if*/
   }/*if*/
   else
    {
     strcpy(buf, program_root_dir);
     strcat(buf, "\\");
     strcat(buf, name_list[c-1].str);
     chdir(buf);

     main_menu();
     textattr(reverse_color);
     clrscr();
    }/*else*/

}

void main(void)
{
```

```
  int  item = 0;
  int  c;
  char st_name[NAME_LEN+1]="";
  HWND wh;
  int  num_ts = 5;  /* num items on main menu*/
 #define buflen 80
  char buf[buflen];

 #if !DEBUG
  spawnlp(P_WAIT, "bird", 0);
 #endif

  puts("Trying to initialize");
  WinInitialize();
  puts("Initialized");

  textattr(reverse_color);
  clrscr();
  title_screen();
  puts("Title Screen");
name_list = (struct text_type*) malloc(sizeof(struct text_type) * MAX_STUDENTS);
  text1 = (struct text_type*) malloc(sizeof(struct text_type) * num_ts);
  student.name = (char *) malloc (sizeof(char) * NAME_LEN);
  student.age = (char *) malloc (sizeof(char) * AGE_LEN);
  student.diagnosis = (char *) malloc (sizeof(char) * DIAG_LEN);
  student.person_1 = (char *) malloc (sizeof(char) * NAME_LEN);
  student.person_2 = (char *) malloc (sizeof(char) * NAME_LEN);
  student.location = (char *) malloc(sizeof(char) * DIAG_LEN);

  for (c=0;c<MAX_STUDENTS;c++)
     name_list[c].str = (char*)malloc(sizeof(char) * NAME_LEN);
        /* allocate name list objects*/

  assign_menu_attribs();
  clear_student_space();

  while (item < num_ts )
  {
   textattr(reverse_color);
   clrscr();
   gotoxy(1,1);
   cprintf("%s   ", getcwd(buf, buflen));

   program_root_dir = (char *) malloc ( sizeof(char) * (strlen(buf)+1) );
   strcpy(program_root_dir, buf);

   open_and_print_student_window(&wh, num_ts);
   item = point_and_shoot(wh, text1, num_ts);
   WinDestroyWindow(wh);

   switch(item)
    {
     case -1: item = num_ts;
           break;
     case  1: strcpy(st_name, add_or_edit_student(ADD_STUDENT, ""));
           break;
```

```
    case  2: delete_or_edit_student(DELETE_STUDENT);
             break;
    case  3: strcpy(st_name, delete_or_edit_student(EDIT_STUDENT));
             break;
    case  4: select_student("");
             break;

   }/*switch*/

  if (strcmp(st_name, ""))
   {
     select_student(st_name);
     strcpy(st_name, "");
   }/*if*/

   free(program_root_dir);
 }/*while*/

 WinDestroyWindow(wh);
 WinTerminate();

 free_all(num_ts);
 flushall();
}
```

```
#include<process.h>
#include<string.h>
#include<stdio.h>
#include<alloc.h>
#include<conio.h>
#include<dir.h>
#include<win.h>
#include<pointsht.h>
#include<keyboard.h>
#include"student.h"

#define REPORT "Report"
#define STS  "STS"
#define SETUP  "Setup"
#define SPEECH_MGR "Speech Manager"
#define EXIT "Exit"
#define MAX_STUDENTS   20     /* max number of students*/
#define NAME_LEN 9
#define AGE_LEN 3
#define DIAG_LEN 21

extern short warning_color;
extern short win_color;
extern short bdr_color;
extern short EXPLODE;

struct text_type *text;
extern struct text_type *name_list;
struct student_type{
  char  *name;
  char  *age;
  char  *diagnosis;
  char  *person_1;
  char  *person_2;
 };

extern struct student_type student;
extern char * program_root_dir;

void wtext_out(HWND wh, int col, int row, char *str, int color);

/*********************************************************************/
/* FUNCTION   open_and_print_main_menu                             */
/* REMARKS    opens and prints the main window of the student       */
/*            program                                               */
/* INPUT      HWND *wh - pointer to the window handle               */
/*            int num_ts - number of menu items                     */
/* OUTPUT                                                           */
/* CALLED BY  main                                                  */
/* CALLS      wtext_out                                             */
/*********************************************************************/

void open_and_print_main_menu(HWND *wh, int num_ts)
{
int c;
```

```
*wh = WinCreateWindow(7, 15, 50, 12, win_color, DBL_LINE_ALL_SIDES, bdr_color, EXPLODE);
  WinCenterText(*wh, 0, "Sound to Speech", win_color);
  for (c=0;c<num_ts;c++)
    wtext_out(*wh, text[c].col, text[c].row, text[c].str, win_color);

}

/**********************************************************************/
/* FUNCTION   assign_central_menu_attribs                           */
/* REMARKS    assigns the menu attributes for the main menu to      */
/*            allow for point and shoot routine                     */
/* INPUT                                                            */
/* OUTPUT                                                           */
/* CALLED BY  main                                                  */
/* CALLS      allocate_text_object                                  */
/**********************************************************************/

void assign_central_menu_attribs(void)
{
int lcol = 2;
int first_row=2;

 allocate_text_object(text, 0, STS, lcol, first_row);
 allocate_text_object(text, 1, SETUP, lcol, first_row+2);
 allocate_text_object(text, 2, SPEECH_MGR, lcol, first_row+4);
 allocate_text_object(text, 3, REPORT, lcol, first_row+6);
 allocate_text_object(text, 4, EXIT, lcol, first_row+8);

}

/**********************************************************************/
/* FUNCTION   speech_mgr                                            */
/* REMARKS    prompts the user to choose between Recognition        */
/*            functions and Recording Functions.                    */
/* INPUT                                                            */
/* OUTPUT                                                           */
/* CALLED BY  main_menu                                             */
/* CALLS      allocate_text_object                                  */
/**********************************************************************/


void speech_mgr(void)
{
int k=0;
struct text_type *text;
HWND wh;
int c=0;

#define RECOGNITION  "Recognition Functions"
#define RECORDING    "Recording Functions"


 text = (struct text_type*) malloc(sizeof(struct text_type) * 3);
 allocate_text_object(text, 0, RECOGNITION, 2, 2);
 allocate_text_object(text, 1, RECORDING, 2, 4);
 allocate_text_object(text, 2, EXIT, 2, 7);
```

```
wh = WinCreateWindow(10, 27, 26, 9, win_color, DBL_LINE_ALL_SIDES, bdr_color, EXPLODE);
  WinCenterText(wh, 0, "Speech Manager", win_color);
  for (c=0;c<=2;c++)
    wtext_out(wh, text[c].col, text[c].row, text[c].str, win_color);

  k = point_and_shoot(wh, text, 3);
  WinDestroyWindow(wh);

  for (c=0;c<=2;c++)
    free(text[c].str);
  free(text);
  switch(k)
   {
      case 1: status(spawnlp(P_WAIT, "strain", 0));
              break;
      case 2: status(spawnlp(P_WAIT, "speech", 0));
              break;

   }/*switch*/

 }
 status(int val)
 {
    if(val ==-1)
      cprintf("failed to start child process\n");
    else
      if(val > 0) cprintf("child terminated abnormally\n");
    delay(5000);
 }
 /*********************************************************************/
 /* FUNCTION   main_menu                                              */
 /* REMARKS    main menu for student program.  From here you can      */
 /*            run STSMAIN, SETUP, SPEECH MANAGER, or return to the   */
 /*            main system menu                                       */
 /* INPUT                                                             */
 /* OUTPUT                                                            */
 /* CALLED BY main                                                    */
 /* CALLS      assign_central_menu_attribs, open_and_print_main_menu */
 /*            speech_mgr                                             */
 /*********************************************************************/


 void main_menu(void)
 {
  char buf[80];
  int buflen=80;
  HWND wh;
  int  item=0, num_ts = 5;
  extern short reverse_color;

  text = (struct text_type*) malloc(sizeof(struct text_type) * 5);

  assign_central_menu_attribs();
  while (item < num_ts && item != -1 )
  {
```

```
      textattr(reverse_color);
      clrscr();

      gotoxy(1,1);
      cprintf("%s   ", getcwd(buf, buflen));

      open_and_print_main_menu(&wh, num_ts);
      item = point_and_shoot(wh, text, num_ts);
      WinDestroyWindow(wh);

      switch(item)
       {
        case -1: item = num_ts;
              break;
        case  1: spawnlp(P_WAIT, "stsmain", 0);
              break;
        case  2: spawnlp(P_WAIT, "setup", 0);
              break;
        case  3: speech_mgr();
               break;
        case  4: spawnlp(P_WAIT, "report", 0);
              break;
       }/*switch*/

    }/*while*/

  chdir(program_root_dir);
  for(item=0;item<=4;item++)
    free(text[item].str);
  free(text);
  }


  /*********************************************************************/
  /* FUNCTION    title_screen()                                       */
  /* REMARKS     explodes the title screen                            */
  /* INPUT                                                            */
  /* OUTPUT                                                           */
  /* CALLED BY   main                                                 */
  /* CALLS       get_key()                                            */
  /*********************************************************************/

  void title_screen(void)
  {
   HWND wh;
   char ver[5]="", buf[14]="Version ";
   FILE *fp;

 wh = WinCreateWindow(5, 10, 60, 15, win_color, DBL_LINE_ALL_SIDES, bdr_color, EXPLODE);
   WinCenterText(wh, 1, "Sound To Speech", win_color);
   fp = fopen("version.dat", "rt");
   if (fp)
      fgets(ver, 6, fp);
   fclose(fp);
   strcat(buf, ver);
```

```
/**********************************************************************/
 /* WIN           - Routines which provide windowing functionality     */
 /*                                                                     */
 /*                                                                     */
 /*                                                                     */
/**********************************************************************/
 /*                       Modification Log                              */
/**********************************************************************/
 /* Version   Date    Programmer         -----------  Description  ------------
-- */
 /*                                                                     */
 /* V01.00   112787  Bob Withers  Program intially complete.            */
 /*                                                                     */
 /*                                                                     */
/**********************************************************************/

 #include <stdlib.h>
 #include <stddef.h>
 #include <string.h>
 #include "win.h"

 #define MAX_WINDOWS             20


 struct sWinData
 {
     BYTE              cRow;
     BYTE              cCol;
     BYTE              cWidth;
     BYTE              cHeight;
     BYTE              cWinWidth;
     BYTE              cWinHeight;
     BYTE              cWinClr;
     BYTE              cBdrType;
     BYTE              cBdrClr;
     BYTE              cCurRow;
     BYTE              cCurCol;
     char             *pHidden;
     char              cSaveData[1];
 };
 typedef struct sWinData WINDATA;
 typedef struct sWinData *PWINDATA;

 static PWINDATA      WinHandle[MAX_WINDOWS + 1];


/**********************************************************************/
 /*  WinCvtHandle    - Convert a window handle into a pointer to the   */
 /*                  window information data structure.                */
 /*  Parms:                                                             */
 /*    hWnd        - handle to the window                              */
 /*                                                                     */
 /*  Return Value:    pointer to WINDATA or NULL if invalid handle     */
/**********************************************************************/

 static PWINDATA near pascal WinCvtHandle(hWnd)
```

```
HWND    hWnd;
{
    if (hWnd < 0 || hWnd > MAX_WINDOWS)
     return(NULL);
    return(WinHandle[hWnd]);
}


/***************************************************************************/
/*  WinGetHandle    - Return an unused window handle.                     */
/*                                                                        */
/*  Parms:          None                                                  */
/*                                                                        */
/*  Return Value:       Window handle or NULL if none available           */
/***************************************************************************/

static HWND near pascal WinGetHandle()
{
    register int      i;

    for (i = 1; i <= MAX_WINDOWS; ++i)
    {
     if (NULL == WinHandle[i])
         return(i);
    }
    return(NULL);
}


/***************************************************************************/
/*  WinExplodeWindow - Draws an exploded window on the screen.            */
/*                                                                        */
/*  Parms:                                                                */
/*    nRow        - Top row of requested window (1 relative)              */
/*    nCol        - Left column of requested window (1 relative)          */
/*    nWidth      - Width (in columns) of requested window                */
/*    nHeight             - Height (in rows) of requested window          */
/*    nWinClr             - Color of the window background                */
/*    nBdrType            - Type of border for this window (defined in WIN.H)
*/
/*                      NO_BOX                                            */
/*                      DBL_LINE_TOP_BOTTOM                               */
/*                      DBL_LINE_SIDES                                    */
/*                      DBL_LINE_ALL_SIDES                                */
/*                      SNGL_LINE_ALL_SIDES                               */
/*                      GRAPHIC_BOX                                       */
/*                      NO_WIND_BORDER                                    */
/*    nBdrClr             - Color or the window border                    */
/*                                                                        */
/*  Return Value:       None                                              */
/***************************************************************************/

void near pascal WinExplodeWindow(nRow, nCol, nWidth, nHeight,
                               nWinClr, nBdrType, nBdrClr)
short           nRow, nCol, nWidth, nHeight;
short           nWinClr, nBdrType, nBdrClr;
{
```

```
    register short      nLRR, nLRC, nX1, nY1, nX2, nY2;

    nLRR   = nRow + nHeight - 1;
    nLRC   = nCol + nWidth - 1;
    nX1    = (nRow + (nHeight >> 1)) - 1;
    nY1    = (nCol + (nWidth >> 1)) - 3;
    nX2    = nX1 + 2;
    nY2    = nY1 + 5;
    while (TRUE)
    {
     ScrClearRect(nX1, nY1, nY2 - nY1 + 1, nX2 - nX1 + 1, nWinClr);
     ScrDrawRect(nX1, nY1, nY2 - nY1 + 1, nX2 - nX1 + 1,
                 nBdrClr, nBdrType);
     if (nX1 == nRow && nY1 == nCol && nX2 == nLRR && nY2 == nLRC)
         break;
     nX1 = (nX1 - 1 < nRow) ? nRow : nX1 - 1;
     nY1 = (nY1 - 3 < nCol) ? nCol : nY1 - 3;
     nX2 = (nX2 + 1 > nLRR) ? nLRR : nX2 + 1;
     nY2 = (nY2 + 3 > nLRC) ? nLRC : nY2 + 3;
    }
    return;
}



/****************************************************************************/
/* WinDrawWindow   - Draws a window on the screen without creating the    */
 /*                    WINDATA structure or saving the previous screen    */
 /*                    contents.                                          */
 /*                                                                       */
 /*  Parms:                                                               */
 /*    nRow        - Top row of requested window (1 relative)             */
 /*    nCol        - Left column of requested window (1 relative)         */
 /*    nWidth      - Width (in columns) of requested window               */
 /*    nHeight            - Height (in rows) of requested window          */
 /*    nWinClr            - Color of the window background                */
 /*    nBdrType           - Type of border for this window (defined in WIN.H)
 */
 /*                        NO_BOX                                         */
 /*                        DBL_LINE_TOP_BOTTOM                            */
 /*                        DBL_LINE_SIDES                                 */
 /*                        DBL_LINE_ALL_SIDES                             */
 /*                        SNGL_LINE_ALL_SIDES                            */
 /*                        GRAPHIC_BOX                                    */
 /*                        NO_WIND_BORDER                                 */
 /*    nBdrClr            - Color or the window border                    */
 /*    bExplodeWin   - Boolean value requesting either a pop-up or        */
 /*                    exploding window                                   */
 /*                        TRUE   ==> Exploding window                    */
 /*                        FALSE  ==> Pop-up window                       */
 /*                                                                       */
 /*  Return Value:     None                                               */
/****************************************************************************/

void pascal WinDrawWindow(nRow, nCol, nWidth, nHeight,
                          nWinClr, nBdrType, nBdrClr, bExplodeWin)
short           nRow, nCol, nWidth, nHeight;
```

```
short           nWinClr, nBdrType, nBdrClr;
short           bExplodeWin;
{
    if (bExplodeWin)
     WinExplodeWindow(nRow, nCol, nWidth, nHeight,
                      nWinClr, nBdrType, nBdrClr);
    else
    {
     ScrClearRect(nRow, nCol, nWidth, nHeight, nWinClr);
     ScrDrawRect(nRow, nCol, nWidth, nHeight, nBdrClr, nBdrType);
    }
    return;
}


/*****************************************************************************/
/* WinCreateWindow - Create a window with the requested attributes and      */
/*                  return a handle which may be used to identify this     */
/*                  particular window in future calls.                      */
/*  Parms:                                                                   */
/*    nRow      - Top row of requested window (1 relative)                  */
/*    nCol      - Left column of requested window (1 relative)              */
/*    nWidth    - Width (in columns) of requested window                    */
/*    nHeight       - Height (in rows) of requested window                  */
/*    nWinClr       - Color of the window background                        */
/*    nBdrType      - Type of border for this window (defined in WIN.H)
*/
/*                    NO_BOX                                                 */
/*                    DBL_LINE_TOP_BOTTOM                                    */
/*                    DBL_LINE_SIDES                                         */
/*                    DBL_LINE_ALL_SIDES                                     */
/*                    SNGL_LINE_ALL_SIDES                                    */
/*                    GRAPHIC_BOX                                            */
/*                    NO_WIND_BORDER                                         */
/*    nBdrClr       - Color for the window border                           */
/*    bExplodeWin  - Boolean value requesting either a pop-up or           */
/*                  exploding window                                        */
/*                    TRUE   ==> Exploding window                           */
/*                    FALSE  ==> Pop-up window                              */
/*                                                                           */
/*  Return Value:                                                            */
/*    hWnd      - Handle of the created window or NULL if an error         */
/*                  prevented the creation                                  */
/*****************************************************************************/

HWND pascal WinCreateWindow(nRow, nCol, nWidth, nHeight,
                            nWinClr, nBdrType, nBdrClr, bExplodeWin)
short           nRow, nCol, nWidth, nHeight;
short           nWinClr, nBdrType, nBdrClr;
short           bExplodeWin;
{
    register PWINDATA           pWinData;
    auto     HWND         hWnd;

    hWnd = WinGetHandle();
    if (NULL == hWnd)
     return(hWnd);
```

```
    pWinData = (PWINDATA) malloc(sizeof(WINDATA)
                          + ScrGetRectSize(nWidth, nHeight));
    if ((PWINDATA) NULL != pWinData)
    {
     WinHandle[hWnd]     = pWinData;
     pWinData->cRow      = (BYTE) nRow;
     pWinData->cCol      = (BYTE) nCol;
     pWinData->cWidth    = pWinData->cWinWidth  = (BYTE) nWidth;
     pWinData->cHeight   = pWinData->cWinHeight = (BYTE) nHeight;
     pWinData->cWinClr   = (BYTE) nWinClr;
     pWinData->cBdrType  = (BYTE) nBdrType;
     pWinData->cBdrClr   = (BYTE) nBdrClr;
     pWinData->cCurRow   = pWinData->cCurCol = (BYTE) 1;
     pWinData->pHidden   = NULL;
     if (NO_WIND_BORDER != nBdrType)
     {
         pWinData->cWinWidth  -= 2;
         pWinData->cWinHeight -= 2;
     }
     ScrSaveRect(nRow, nCol, nWidth, nHeight, pWinData->cSaveData);
     if (bExplodeWin)
         WinExplodeWindow(nRow, nCol, nWidth, nHeight,
                          nWinClr, nBdrType, nBdrClr);
     else
     {
         ScrClearRect(nRow, nCol, nWidth, nHeight, nWinClr);
         ScrDrawRect(nRow, nCol, nWidth, nHeight, nBdrClr, nBdrType);
     }
     WinSetCursorPos((HWND) pWinData, 1, 1);
    }
    return(hWnd);
}


/****************************************************************************/
/* WinDestoryWindow - Destroy the window represented by hWnd and replace   */
/*                    the previous screen contents saved when the window   */
/*                    was created.                                          */
/*  Parms:                                                                  */
/*    hWnd        - Handle to the window to be destroyed                    */
/*                                                                          */
/* Return Value:    TRUE => window destroyed, FALSE => invalid handle       */
/****************************************************************************/

BOOL pascal WinDestroyWindow(hWnd)
HWND    hWnd;
{
    register PWINDATA            pWinData;

    pWinData = WinCvtHandle(hWnd);
    if (NULL == pWinData)
     return(FALSE);
    ScrRestoreRect(pWinData->cRow, pWinData->cCol, pWinData->cWidth,
                   pWinData->cHeight, pWinData->cSaveData);
    if (NULL != pWinData->pHidden)
     free(pWinData->pHidden);
```

```
    free((char *) pWinData);
    WinHandle[hWnd] = NULL;
    return(TRUE);
}


/*************************************************************************/
/*   WinScrollWindowUp  - Scrolls the requested window up one line.        */
/*                                                                         */
/*   Parms:                                                                */
/*     hWnd        - Handle to the window to be scrolled                   */
/*                                                                         */
/*   Return Value:      None                                               */
/*************************************************************************/

void pascal WinScrollWindowUp(hWnd)
HWND      hWnd;
{
    register PWINDATA            pWinData;
    auto     short       nRow, nCol;

    pWinData = WinCvtHandle(hWnd);
    if (NULL == pWinData)
     return;
    if (NULL == pWinData->pHidden)
    {
     nRow = pWinData->cRow;
     nCol = pWinData->cCol;
     if (NO_WIND_BORDER != pWinData->cBdrType)
     {
         nRow++;
         nCol++;
     }
     ScrScrollRectUp(nRow, nCol, pWinData->cWinWidth,
                     pWinData->cWinHeight, 1, pWinData->cWinClr);
    }
    return;
}


/*************************************************************************/
/*   WinScrollWindowDown - Scrolls the requested window down one line.     */
/*                                                                         */
/*   Parms:                                                                */
/*     hWnd        - Handle to the window to be scrolled                   */
/*                                                                         */
/*   Return Value:      None                                               */
/*************************************************************************/

void pascal WinScrollWindowDown(hWnd)
HWND      hWnd;
{
    register PWINDATA            pWinData;
    auto     short       nRow, nCol;

    pWinData = WinCvtHandle(hWnd);
```

```
    if (NULL == pWinData)
     return;
    if (NULL == pWinData->pHidden)
    {
     nRow = pWinData->cRow;
     nCol = pWinData->cCol;
     if (NO_WIND_BORDER != pWinData->cBdrType)
     {
         nRow++;
         nCol++;
     }
     ScrScrollRectDown(nRow, nCol, pWinData->cWinWidth,
                       pWinData->cWinHeight, 1, pWinData->cWinClr);
    }
    return;
}


/*****************************************************************************/
/* WinSetCursorPos - Position the cursor relative to the selected window. */
 /*                 The upper left hand corner of the window is (1,1)       */
 /*                                                                         */
 /*  Parms:                                                                 */
 /*    hWnd        - Handle to the window to position the cusor in         */
 /*    nRow        - Row to position cursor to within window (1 relative)  */
 /*    nCol        - Col to position cursor to within window (1 relative)  */
 /*                                                                         */
 /*  Return Value:    None                                                  */
/*****************************************************************************/

 void pascal WinSetCursorPos(hWnd, nRow, nCol)
 HWND         hWnd;
 short                nRow, nCol;
 {
     register PWINDATA          pWinData;
     auto     short        nMaxRow, nMaxCol;

     if (NULL == hWnd)
     {
      ScrSetCursorPos(nRow, nCol);
      return;
     }
     pWinData = WinCvtHandle(hWnd);
     if (NULL == pWinData)
      return;
     if (nRow > pWinData->cWinHeight && nCol > pWinData->cWinWidth)
      return;
     pWinData->cCurRow = (BYTE) nRow;
     pWinData->cCurCol = (BYTE) nCol;
     nRow = nRow + pWinData->cRow - 1;
     nCol = nCol + pWinData->cCol - 1;
     if (NO_WIND_BORDER != pWinData->cBdrType)
     {
      ++nRow;
      ++nCol;
     }
```

```
        ScrSetCursorPos(nRow, nCol);
        return;
    }


/*****************************************************************************/
 /*   WinClearScreen - Clear a window to the desired color.                   */
 /*                                                                           */
 /*   Parms:                                                                  */
 /*     hWnd       - Handle to the window to be cleared                       */
 /*                  (A handle of NULL clears the entire screen)              */
 /*     nColor     - Color to be used in clearing the window                  */
 /*                                                                           */
 /*   Return Value:    None                                                   */
/*****************************************************************************/

 void pascal WinClearScreen(hWnd, nColor)
 HWND     hWnd;
 short            nColor;
 {
     register PWINDATA          pWinData;
     auto     short       nRow, nCol;

     if (NULL == hWnd)
      ScrClearRect(1, 1, 80, 25, nColor);
     else
     {
      pWinData = WinCvtHandle(hWnd);
      if (NULL == pWinData)
          return;
      nRow     = pWinData->cRow;
      nCol     = pWinData->cCol;
      if (NO_WIND_BORDER != pWinData->cBdrType)
      {
          ++nRow;
          ++nCol;
      }
      pWinData->cWinClr = (BYTE) nColor;
      ScrClearRect(nRow, nCol, pWinData->cWinWidth, pWinData->cWinHeight,
                   pWinData->cWinClr);
     }
     return;
 }


/*****************************************************************************/
 /*   WinTextOut          - Display a string to the requested window at the
/
 /*                        current cursor location (for that window) using the  */
 /*                        passed color attribute.                              */
 /*                        If the string extends beyond the boundries of the    */
 /*                        window it will be truncated.                         */
 /*   Parms:                                                                    */
 /*     hWnd      - Handle of the window                                        */
 /*     pStr      - Pointer to the NULL terminated string to display            */
 /*     nAttr     - Color attribute to be used in displaying the string         */
 /*                                                                             */
```

```
/*  Return Value:    None                                               */
/************************************************************************/

void pascal WinTextOut(hWnd, pStr, nAttr)
HWND     hWnd;
char    *pStr;
short            nAttr;
{
    register PWINDATA        pWinData;
    auto     short   nCount;
    auto     short   nRow, nCol;

    pWinData = WinCvtHandle(hWnd);
    if (NULL == pWinData)
     return;
    ScrGetCursorPos(&nRow, &nCol);
    WinSetCursorPos(hWnd, pWinData->cCurRow, pWinData->cCurCol);
    nCount = pWinData->cWinWidth - pWinData->cCurCol + 1;
    ScrTextOut(pStr, nAttr, nCount);
    ScrSetCursorPos(nRow, nCol);
    return;
}


/************************************************************************/
/*  WinCenterText  - Centers a text string in a window.                 */
/*                                                                      */
/*  Parms:                                                              */
/*    hWnd       - Handle of the window                                 */
/*    nRow       - Window row to place the string on                    */
/*    pStr       - Pointer to the string to be displayed                */
/*    nColor     - Color attribute used to display the string           */
/*                                                                      */
/*  Return Value:    None                                               */
/************************************************************************/
void pascal WinCenterText(hWnd, nRow, pStr, nColor)
HWND     hWnd;
short            nRow;
char    *pStr;
short            nColor;
{
    if (NULL == WinCvtHandle(hWnd))
     return;
 WinSetCursorPos(hWnd, nRow, (WinGetWindowWidth(hWnd) - strlen(pStr)) / 2);
    WinTextOut(hWnd, pStr, nColor);
    return;
}


/************************************************************************/
/*  WinMoveWindow  - Move an existing window to a new screen location.   */
/*                   In this version the window to be moved MUST be fully */
/*                   visible on the screen for WinMoveWindow to perform   */
/*                   properly.  If the window being moved is completely or */
/*                   partially under another window the screen will not    */
/*                   be left in the correct state (i.e. garbage on screen).*/
```

```
/*                      It is the callers responsibility to insure that the   */
/*                      window is not being moved off the screen.  Even with  */
/*                      these restriction this can be a handy routine and is  */
/*                      included for that reason.  A future release of the    */
/*                      package may fix these shortcomings.                   */
/*                                                                            */
/*  Parms:                                                                    */
/*    hWnd         - Handle to the window to be moved                        */
/*    nRow         - Move the window to this row                             */
/*    nCol         - Move the window to this column                          */
/*                                                                            */
/*  Return Value:     None                                                    */
/****************************************************************************/

void pascal WinMoveWindow(hWnd, nRow, nCol)
HWND     hWnd;
short              nRow, nCol;
{
    register PWINDATA            pWinData;
    register char       *pBuf;

    pWinData = WinCvtHandle(hWnd);
    if (NULL == pWinData)
     return;
    if (NULL != pWinData->pHidden)
    {
     pWinData->cRow = (BYTE) nRow;
     pWinData->cCol = (BYTE) nCol;
     return;
    }
    pBuf      = malloc(ScrGetRectSize(pWinData->cWidth, pWinData->cHeight));
    if (NULL != pBuf)
    {
     ScrSaveRect(pWinData->cRow, pWinData->cCol,
                 pWinData->cWidth, pWinData->cHeight, pBuf);
     ScrRestoreRect(pWinData->cRow, pWinData->cCol,
                    pWinData->cWidth, pWinData->cHeight,
                    pWinData->cSaveData);
     ScrSaveRect(nRow, nCol, pWinData->cWidth, pWinData->cHeight,
                 pWinData->cSaveData);
     ScrRestoreRect(nRow, nCol, pWinData->cWidth, pWinData->cHeight, pBuf);
     pWinData->cRow = (BYTE) nRow;
     pWinData->cCol = (BYTE) nCol;
     free(pBuf);
    }
    return;
}


/****************************************************************************/
/* WinGetWindowRow - Returns the row value currently associated with the    */
/*                      passed window handle.                                 */
/*  Parms:                                                                    */
/*    hWnd         - Handle to the window                                     */
/*                                                                            */
/*  Return Value:     Row the window currently resides at                     */
```

```
/*****************************************************************************/

short pascal WinGetWindowRow(hWnd)
HWND      hWnd;
{
    register PWINDATA           pWinData;

    pWinData = WinCvtHandle(hWnd);
    if (NULL == pWinData)
     return(0);
    return(pWinData->cRow);
}


/*****************************************************************************/
/*  WinGetWindowCol - Returns the col value currently associated with the   */
 /*                    passed window handle.                                 */
 /*  Parms:                                                                  */
 /*    hWnd       - Handle to the window                                     */
 /*                                                                          */
 /*  Return Value:    Column the window currently resides at                 */
/*****************************************************************************/

short pascal WinGetWindowCol(hWnd)
HWND      hWnd;
{
    register PWINDATA           pWinData;

    pWinData = WinCvtHandle(hWnd);
    if (NULL == pWinData)
     return(0);
    return(pWinData->cCol);
}


/*****************************************************************************/
 /*  WinGetWindowWidth - Returns the column width of the passed window.      */
 /*                                                                          */
 /*  Parms:                                                                  */
 /*    hWnd       - Handle to the window                                     */
 /*                                                                          */
 /*  Return Value:    Number of columns in the window                        */
/*****************************************************************************/

short pascal WinGetWindowWidth(hWnd)
HWND      hWnd;
{
    register PWINDATA           pWinData;

    pWinData = WinCvtHandle(hWnd);
    if (NULL == pWinData)
     return(0);
    return(pWinData->cWinWidth);
}
```

```
/***************************************************************************/
/* WinGetWindowHeight - Returns the number of rows in the passed window.  */
/*                                                                         */
/*  Parms:                                                                 */
/*    hWnd        - Handle to the window                                   */
/*                                                                         */
/*  Return Value:    Number of rows in the window                          */
/***************************************************************************/

short pascal WinGetWindowHeight(hWnd)
HWND     hWnd;
{
    register PWINDATA            pWinData;

    pWinData = WinCvtHandle(hWnd);
    if (NULL == pWinData)
     return(0);
    return(pWinData->cWinHeight);
}


/***************************************************************************/
/*  WinGetWindowClr - Get the window background color                      */
/*                                                                         */
/*  Parms:                                                                 */
/*    hWnd        - Handle to the window                                   */
/*                                                                         */
/*  Return Value:    Returns the attribute for the window color            */
/***************************************************************************/

short pascal WinGetWindowClr(hWnd)
HWND     hWnd;
{
    register PWINDATA            pWinData;

    pWinData = WinCvtHandle(hWnd);
    if (NULL == pWinData)
     return(0);
    return(pWinData->cWinClr);
}


/***************************************************************************/
/*  WinGetWindowBdrClr - Get the window border color                       */
/*                                                                         */
/*  Parms:                                                                 */
/*    hWnd        - Handle to the window                                   */
/*                                                                         */
/*  Return Value:    Returns the attribute for the window border color     */
/***************************************************************************/

short pascal WinGetWindowBdrClr(hWnd)
HWND     hWnd;
{
    register PWINDATA            pWinData;
```

```
        pWinData = WinCvtHandle(hWnd);
        if (NULL == pWinData)
         return(0);
        return(pWinData->cBdrClr);
    }


/*************************************************************************/
/*   WinGetBorderType - Gets the border type of the passed window       */
/*                                                                       */
/*   Parms:                                                              */
/*     hWnd       - Handle to the window                                 */
/*                                                                       */
/*   Return Value:    Returns the window border type                     */
/*************************************************************************/

    short pascal WinGetBorderType(hWnd)
    HWND      hWnd;
    {
        register PWINDATA            pWinData;

        pWinData = WinCvtHandle(hWnd);
        if (NULL == pWinData)
         return(0);
        return(pWinData->cBdrType);
    }


/*************************************************************************/
/*   WinHideWindow  - Removes a window from the screen, saving it's     */
/*                    contents.  The window can later be placed back on  */
/*                    the screen via WinShowWindow().  Note that in this */
/*                    release the window MUST be fully visible for this  */
/*                    operating to work correctly.                       */
/*   Parms:                                                              */
/*     hWnd       - Handle to the window                                 */
/*                                                                       */
/*   Return Value:    TRUE => window hidden, FALSE => buf alloc failed   */
/*************************************************************************/

    BOOL pascal WinHideWindow(hWnd)
    HWND      hWnd;
    {
        register PWINDATA           pWinData;
        auto     char       *pBuf;
        auto     short       nBufSize;
        auto     short       nRow, nCol, nWidth, nHeight;

        pWinData = WinCvtHandle(hWnd);
        if (NULL == pWinData)
         return(FALSE);
        nRow     = pWinData->cRow;
        nCol     = pWinData->cCol;
        nWidth   = pWinData->cWidth;
        nHeight  = pWinData->cHeight;
        nBufSize = ScrGetRectSize(nWidth, nHeight);
```

```
     if (NULL != pWinData->pHidden)
      free(pWinData->pHidden);
     pBuf      = malloc(nBufSize);
     if (NULL == pBuf)
      return(FALSE);
     ScrSaveRect(nRow, nCol, nWidth, nHeight, pBuf);
     ScrRestoreRect(nRow, nCol, nWidth, nHeight, pWinData->cSaveData);
     pWinData->pHidden = pBuf;
     return(TRUE);
 }


/*************************************************************************/
/* WinShowWindow  - Places a hidden window back on the screen and frees  */
 /*                   the buffer used to hold the window image.          */
 /*  Parms:                                                              */
 /*    hWnd       - Handle to the window                                 */
 /*                                                                      */
/* Return Value:   TRUE => window shown, FALSE => window wasn't hidden   */
/*************************************************************************/

 BOOL pascal WinShowWindow(hWnd)
 HWND      hWnd;
 {
     register PWINDATA            pWinData;

     pWinData = WinCvtHandle(hWnd);
     if (NULL == pWinData)
      return(FALSE);
     if (NULL == pWinData->pHidden)
      return(FALSE);
     ScrRestoreRect(pWinData->cRow, pWinData->cCol, pWinData->cWidth,
                 pWinData->cHeight, pWinData->pHidden);
     free(pWinData->pHidden);
     pWinData->pHidden = NULL;
     return(TRUE);
 }


/*************************************************************************/
 /*  WinInitialize  - Init the windowing system.                         */
 /*                                                                      */
 /*  Parms:         None                                                 */
 /*                                                                      */
 /*  Return Value:    None                                               */
/*************************************************************************/

 void pascal WinInitialize()
 {
     ScrInitialize();
     memset((char *) WinHandle, NULL, sizeof(WinHandle));
     return;
 }


/*************************************************************************/
```

```
/*  WinTerminate   - Clean up the windowing package                    */
/*                                                                     */
/*  Parms:          None                                               */
/*                                                                     */
/*  Return Value:    None                                              */
/**********************************************************************/

void pascal WinTerminate()
{
    register short     i;

    for (i = 1; i <= MAX_WINDOWS; ++i)
    {
     if (WinHandle[i] != NULL)
         WinDestroyWindow(i);
    }
    return;
}
```

THE SOUND-TO-SPEECH TRANSLATION SYSTEM UTILIZING PHOTOGRAPHIC-QUALITY GRAPHIC SYMBOLS

Operation and Training Manual

Grant No. H180P90015

TABLE OF CONTENTS

Forward

The ARC's Bioengineering Program is currently involved in research utilizing voice recognition systems designed to allow individuals with mental retardation and/or severe handicaps to control their environment and communicate with others. The most recent development in this research track is the Sound-to-Speech Translations Using Graphics Symbols (STS/Graphics) computer-based system which incorporates a photographic-quality graphics display. The STS/Graphics system accepts keyboard input and delivers environmental control and digitized speech output. Since 1987, the system has undergone three rounds of alpha testing and one round of beta testing resulting in system refinements which increase speed of output and ease of training the system to recognize the user's vocalizations.

The system was developed to enable persons with mental retardation to use their own voices to communicate. Utilizing the photographic quality images with the system turns a simple communication board into a communication system which the user can operate using their own speech.

Funding for the Sound-To-Speech system development was provided through Grant No. H180P90015 from the U.S. Department of Education.

Section 1
Introduction

THE SOUND-TO-SPEECH SYSTEM

What Is It and Who Is It For?

The Sound-To-Speech system is an electronic system which enables persons who have unintelligible speech to more effectively communicate and control their environment. Individuals may be nonspeaking due to a variety of impairments which make their speech difficult for all but familiar listeners or communication partners to understand. Additionally, these persons may experience concomitant physical and/or cognitive differences which further compromise their independence.

Today, there are a variety of commercially available communication aids for persons who may benefit from the use of an augmentative communication system. Traditionally these systems require operation using a switch or by pointing to the display surface. There is an absence of systems which utilize the user's own voice for communication. Voice-recognition systems which are currently available require the user to control them via intelligible speech commands. The product of these commands is typically text production. These systems are mostly widely used by industry or the military. Text is printed on the computer screen or other multi-task steps. These systems present barriers for persons who are unable to read, or are physically and/or cognitively impaired.

The Sound-To-Speech System was designed to enable nonspeaking persons with a variety of abilities to use their own voice to communicate and/or control their environment. Furthermore, the sound-to-speech system is not a test-based system. Communication displays are available which can be customized to meet the user's needs. Photographs are taken of desired items from the user's environment and scanned into the system. These scanned images can then be placed in the user's communication displays and appear as photographic quality images. The system also assists persons with physical impairment to control their environment by enabling them to control simple household appliances (i.e. TV, VCR, radio, lights etc.).

The system is unique for the following reasons:

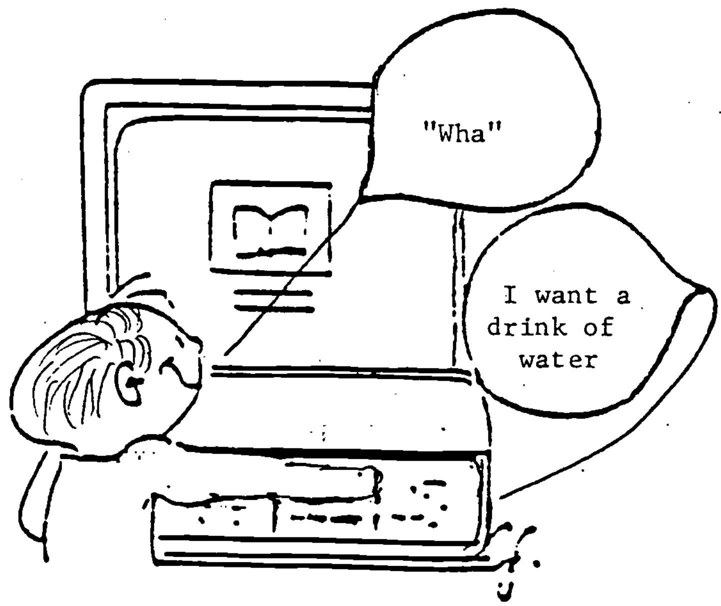
1. Enables the user to use their own voice to communicate

2. The system translates unintelligible speech sounds in to meaningful speech messages which are spoken by the computer

3. Use of photographic quality images on the communication displays which represent choices/activities in the user's natural environment

4. Speech output that is digitized easily understood, age and gender appropriate

5. Data on system use is automatically recorded, thereby facilitating the development of appropriate training strategies


"Wha"
I want a drink of water

System Control

Users can control the system in two ways: **scanning** and **direct selection.**

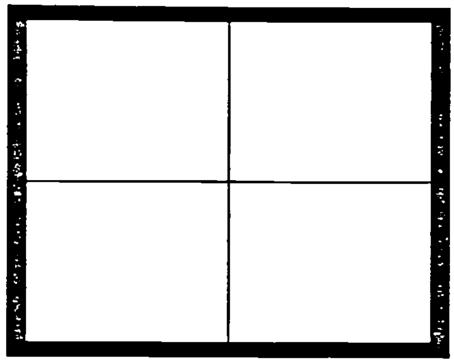
Scanning Using their voice as a switch, the user may stop a highlighted box when it reaches a desired item on the display. Two scanning motions are available in the STS/Graphics software which include linear and row/column scanning.

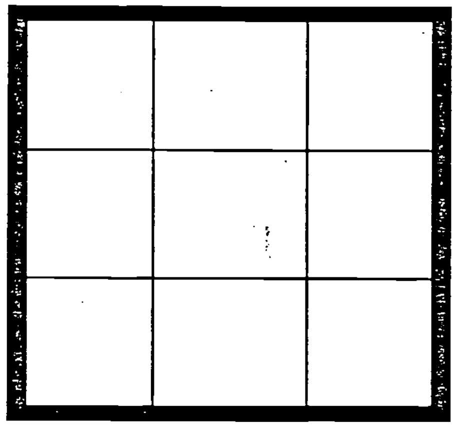
Selection of either mode is based on the user's cognitive and physical abilities. In each case, the system requires a single vocalization. Vocalizations are not required to be consistently similar in their production. Once the scanning action is stopped by the user, the functions associated with that picture image are executed which may include a spoken message and/or environmental control function.

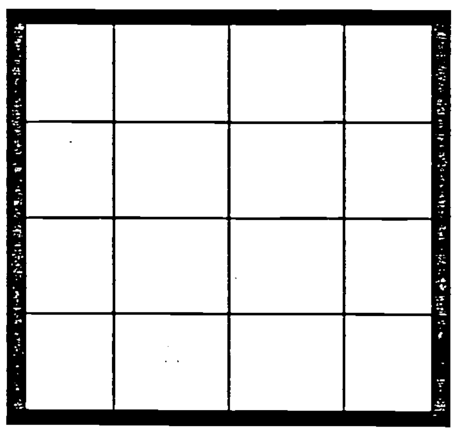
Some individuals are unable to produce a variety of sounds for each message they wish to convey and present physical abilities which prohibit their ability to point to items on a communication display. Using their voice as a switch they are able to look at a communication display and choose the item from the display as it is presented by the computer. They use their own voice as a switch to indicate their choice to the computer which then provides their message and/or activates an appliance in their environment.

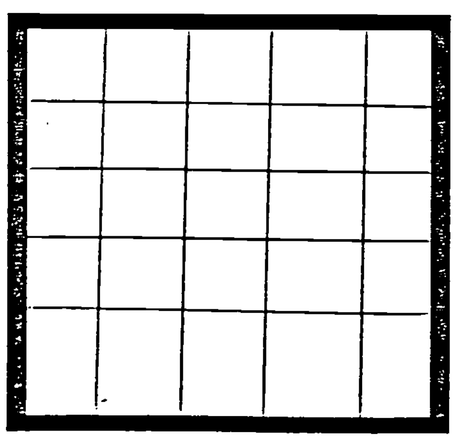
For individuals who require and prefer the use of a switch, the system has been adapted to accept any switch with a 1/8" stereo plug.

Direct selection Using different and distinct, yet unintelligible vocalizations, an individual may select any item appearing on the display. The system is configured to recognize each individual user's "voice". Information is stored for each individual which enables the system to recognize their speech when their display is presented on the monitor. In this mode, the user chooses a sound or vocalization which will be used to retrieve a message from the system. For example, the sound "ma" may mean "I want to listen to music." An image of a radio or tape recorder may appear on the display in front of them. By producing the sound "ma" the system will respond by producing the message "I want to listen to music" and/or activation the radio or tape recorder.









Display Sizes

Displays can be created of varying sizes ranging from 2 x 2 to a 5 x 5 matrix. The size of the display and the number of images used is dependent on the needs and abilities of the user. For each item chosen for the display, a spoken message and/or environmental control function can be selected. For example, a photograph of a tape player might be configured to produce the message "I like this music" as well as activating the tape recorder to play music.

Customizable User Options

A variety of options are available on the system which enhance its customization for the user. They include:

Scan Speed The scan speed in the linear or row/column scanning mode can be adjusted to the user's needs.

Recognition Level the recognition level of the system can be "tuned" to recognize the distinct features of the user's speech sounds. How distinctly these sounds must be produced to be recognized by the system may be adjusted to accommodate the user's abilities.

Matrix and Scan Box Color The color of the display or matrix and the scan box may be changed. Field test studies with the STS system determined that for some users, the contrast between the matrix and scan box color was important to their ability to attend and visually track activity on the display.

Auditory Signal In the scanning mode, the system has the capability to provide a beep or signal as it advances from one image to the next. This signal can be turned on or off while the display is in use. For some of the field test participants the signal was needed to draw their attention to the display as well as assist in visual scanning and tracking activity on the display.

Speech Output

The STS system produces digitized speech messages in response to a user's speech sound. Digitized speech sounds very much like a recording of human speech and can be customized to the user's age and gender. It is highly intelligible and is most effective with user's who present cognitive impairment.

Recordings of the user's messages can be provided by an individual who is similar in age and gender and stored in the system. When the user produces a speech sound, the accompanying recorded message is heard by the communication partner.

INSERT ILLUSTRATION OF REPORT SUMMARY HERE

The Arc
SOUND-TO-SPEECH

STUDENT SUMMARY

Date/Time: Wed Jan 29 09:14:37 1992
Name: Maggie Sauer
Age: -
Diagnosis: -
Assistant 1: -
Assistant 2: -
Location: -

DISPLAY CONFIGURATION

Access Method: Direct Select
Page: DIRECT
Matrix Size: 3 x 3
Initial Recognition Distance: 50
Matrix Color: Black
Scan Color: Green

CELL INFORMATION

Cell 1: Synonym:	Message: 0
Cell 2: LIPGLOSS Synonym: LIPGLOSS	Message: 3
Cell 3: Synonym:	Message: 0
Cell 4: REDPOL Synonym: REDPOL	Message: 1
Cell 5: Synonym:	Message: 0
Cell 6: LOTION Synonym: LOTION	Message: 2
Cell 7: Synonym: STIC	Message: 0
Cell 8: STICKER Synonym: STIC	Message: 4
Cell 9: Synonym: STIC	Message: 0

Data Collection

A chronological report is generated of all system activity whenever the system is used. Subsidiary reports are generated from the chronological report detailing specific activities such as environmental control audio playback, the number of times each selection was made and a summary of the display setup. The subsidiary reports include an analysis of these activities.

This method of data collection assists those providing support to the user and the system to look more carefully at user performance and provide appropriate training strategies. It also allows the user's performance to be documented over time.

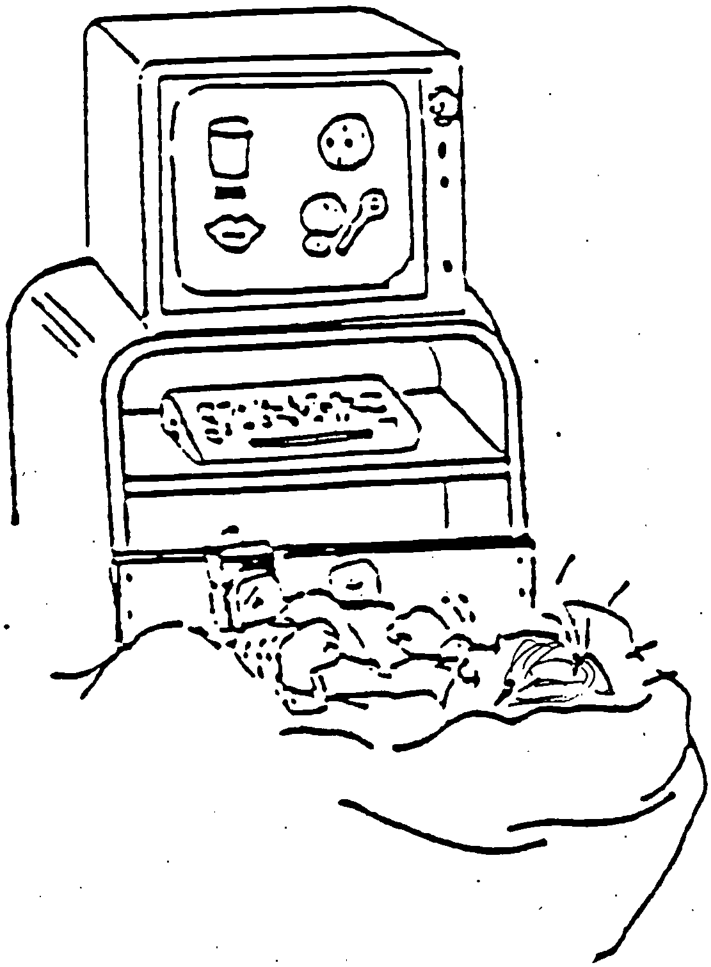
Multiple User Memory

The system has the capability to store communication displays for a variety of users. Thus, the STS system can be customized for one or more users. However, only one person may use the system at one time. The system is not able to recognize the speech sounds of more than one user at a time.

Appropriate Candidates for the STS System

Individuals with a variety of presenting etiologies field tested the STS system. Some of these included cerebral palsy, autism, mental retardation and Down's Syndrome. Most often, these individuals were described as "very talkative" regardless of the intelligibility of their speech. In some cases, a variety of other systems which required them to point to a picture/display or use a switch had been tried. In each case, they preferred to rely on speech and other communication strategies such as gestures, eyegaze, facial expression and interpretation by a familiar communication partner. From these experiences, a list of user characteristics was generated:

1. adequate vision and hearing
2. adequate positioning and seating
3. understanding of picture/object associations
4. possess rudimentary scanning skills using a switch (if appropriate)
5. speech is not functional as a primary mode of communication



Considerations for Use of the STS System

While the considerations outlined below are not meant to preclude a potential user from operation of the STS system, they are provided to suggest areas which can make the difference between optimal and inconsistent performance.

Seating/Positioning Appropriate seating and positioning in a chair, wheelchair etc. is essential to effective system operation. These arrangements will permit the user to best utilize his/her body to control a switch (if appropriate) , body posture for phonation and speech production and monitoring system operation/communication displays.

It is advised that if effective seating and positioning arrangements have not been obtained, an evaluation by a physical and occupational therapist may assist in these determinations. These recommendations can facilitate effective system use and/or switch operation.

Cognitive Considerations The STS system incorporates the use of photographs from the user's natural environment into the system. Additionally, the photographs can be displayed in a variety of sizes. For these reasons, the STS system can accommodate many needs presented by user's with cognitive differences. However, in order to utilize the system most effectively, users should have an understanding of picture/object associations.

For the individual operating the system using the scanning mode, an understanding of cause/effect using a switch is desired.

Speech In the direct selection mode, the user is not required to produce intelligible speech sounds. However, the sounds must be used consistently and repeatable on demand. Some sound production variation can be tolerated by the system, however, as the number of messages in the system increases, the distinctness of the sound is of greater importance.

In the scanning mode, the system will recognize any sound produced by the user and thereby activate the appropriate item on the display. It is not necessary for the speech sound to be consistently produced each time.

Single Switch Control In the scanning mode, the STS system can be controlled using any switch with a 1/8" stereo plug. Thus, for individuals for whom the STS system is appropriate but require use of a single switch, this is possible.

The decision to use a single switch, should be determined by the user with an occupational therapist or other professional who can most effectively determine range of motion and movement. These recommendations should be used in the selection of a switch as well as switch placement.

Microphone Placement A variety of microphones are available which can be used with the STS system. Some individuals prefer not to have a microphone attached to their clothing, mounted in front of their mouth etc. Microphones which best accommodate the needs of the user and their preference can be purchased from manufacturers who sell electronic equipment. Additional information is available in the Appendix.

Vision/Hearing A determination of the user's functional hearing and vision should be established. These abilities enable the user to monitor system activity. With regard to vision, the user must be able to see the display as well as visually track/scan the display for the image of the desired item. This is particularly important for individuals using the system in the scanning mode.

Team Decisions Regarding Candidacy As described above, the information necessary to determine the appropriateness of the system for an individual can be provided by a number of individuals. This includes the user, family, occupational therapist, physical therapist, speech/language pathologist, teacher etc. It is recommended that all these individuals participate in the evaluation and implementation of the system to ensure its effectiveness not only at the time of evaluation but in support of the system later.

Section 2
Software Description

Sound-to-Speech Software Introduction

SYSTEM DESCRIPTION

The main features of the STS/Graphics system are as follows:

Voice Input

The system uses the Votan 2000 voice recognition circuit board and software routines. Each user can store voice templates for sets of up to 64 messages, consisting of sounds or word phrases of up to 8 seconds in duration. Multiple users can simultaneously store and access templates for their messages. The templates for each message can be entered into the system in 2 to 3 passes. For individuals for whom consistent sound production is not possible, the STS/Graphics system can be adapted to accept simple vocalizations as a means of selecting desired items from a visual scanning routine.

Sound-to-Speech Translation and Expansion

The STS/Graphics system incorporates an algorithm that translates any designed sound input, whether intelligible or not, into a specified speech output. For example, if a user's vocalization for water approximates "wuh" and s/he only speaks in single-syllable vocalizations, the system can immediately output "Could I have a drink of water, please." The speech output is digitized and generated by the Votan 2000 circuit board. Any single speech output can be up to 8 seconds in duration, thus permitting single words or complete sentences output.

Environmental Control

The STS/Graphics system incorporates X-10 technology. Any voice (or other) input can be linked to the activation and deactivation of any electrical device. A single voice input can be linked to any combination of spoken outputs and device activations.

Graphics System Interface

The STS/Graphics system is able to generate photographic-quality images of items, appliances or people in the user's environment. Both the size of the image and the number of images appearing on the display can be customized for a user.

SOUND TO SPEECH/GRAPHICS SYSTEM REQUIREMENTS

The following hardware and software components are necessary for operation of the Sound-to-Speech software. It should be noted that the equipment listed below is the minimum configuration required for software operation. Enhanced performance is obtained using a 386 processor and hard disk with additional storage capability.

Hardware and Software Requirements

1. IBM PC-AT
2. 30 MB of Hard Disk Space
3. Super VGA Monitor
4. Paradise Professional Video Adaptor
5. Votan 2000 SRB
6. Image Capturing Device
7. X10 Powerhouse with Serial connector and wall modules
8. 640KB RAM
9. 5 1/4" Floppy drive
10. MS-DOS 3.30
11. Sound to Speech Software
12. Image Capturing Device Driver
13. Image manipulator (PC Paintbrush IV+ Recommended)

Software Configuration and Design

The Sound-to-Speech/Graphics System permits communication displays to be created which are customized to best meet the needs of the user. Photographs are taken of familiar items in the user's environment which can be incorporated into the displays. The system operator uses a scanner to place the image onto a floppy disk. It is then transferred to the display using the STS/Graphics software.

Displays can be created of varying sizes ranging from 2 x 2 to a 5 x 5 matrix. The size of the display and the number of images used is dependent on the needs and abilities of the user. For each item chosen for the display, a spoken message and/or environmental control function can be selected. For example, a photograph of a tape player might be configured to produce the message "I like this music" as well as activating the tape recorder to play music.

Image Capturing

The Sound-to-Speech software is designed specifically to utilize high resolution, color graphics. Using an image capturing device such as a scanner or a camera, photographs of objects are reproduced into a PCX file format which is readable by the STS/Graphics software.

After the photographs have been digitized it is necessary to size them for use in the matrices used by the Sound-to-Speech software. A "paint" style software package is used to size the image to the exact specifications of the Sound to Speech package.

Creating Photo and Message Libraries

Photo Library. Prior to operation of any of these software components, pictures of objects need to be digitized. It is recommended that a collection of photos are made to facilitate speedy operation of the software. Photo images may be stored according to specific topic categories and size. These images are stored on floppy disks for easy retrieval and use during display construction.

Message Library. A second library should be created of audio output messages. These are also categorized into groups which are similar to the library of digitized pictures. ARC has compiled a audio message library for both male and female voices. These messages are stored on floppy disks for easy retrieval and use in constructing user's displays.

MAJOR SOFTWARE COMPONENTS

The Sound-to-Speech software package is made up of three major components: Setup, Speech Manager, and Sound-to-Speech (STS). Each of these components is described briefly in the following pages.

Setup

The Setup section of the STS/Graphics software is responsible for creating a working display for communication and environmental control. The Setup portion of the STS/Graphics software operates in two modes: create a page or edit an existing page. A page is a collection of images and computer controlled functions which correspond with those images. Those images are arranged on a matrix in rows and columns. The dimensions of a matrix, e.g. 4 by 4, defines the number of rows by the number of columns. A cell is one location in the matrix.

Building a Page. The Setup section combines picture files, audio output messages, and/or environmental control commands into designated cells of the matrix or display and facilitates the execution of these commands simultaneously. When information has been entered for each designated cell in the matrix, the information is stored as a "page" file. The number of cells which are filled is determined by the needs of the user.

Setup allows the teacher to define the page name, the matrix dimensions and to place the pictures and recordings from picture and audio libraries into the specified cell of the matrix being built.

At this stage, the user may also define any environmental control functions that are to be included in the page which being created. If an X-10 function is selected from the menu a prompt is provided by the software requesting the user to enter information regarding the wall module code associated with the environmental control function. This information will be assigned to designated cells contained on the matrix. Once a page is selected for use, all specifications determined in the Setup portion of the software are active and page creation is complete.

Speech Manager

Speech Manager provides the audio recording and system training functions for the speech recognition board (SRB).

Audio Recordings. Speech Manager facilitates categorization of individual audio messages into categories appropriate to the user or setting. New categories and messages can quickly and easily be added, similarly unused categories can be deleted. Once a category has been defined, audio output messages may be added, re-recorded or deleted. For each audio output message an area is available to further describe the stored message. These descriptions may be added to, deleted or modified. Message descriptions for a category can be displayed on screen, or printed for future reference.

System Training. Speech Manager processes and stores the utterances or words that are used to control the system through direct selection or scanning modes. For use in direct selection access, the system trainings generated with the user's utterance are matched with the associated picture. Thus the sound "pa" may be associated with a picture of nail polish, and "ray" may be matched to a picture of a radio. For direct selection access, each new picture included in the matrix is represented by a different sound. New sounds may be added to the file or sounds that are not readily recognized by the system may be retrained.

Utilizing the scanning access method a single sound is used to train the system. Once a single sound is recorded and learned by the system, the range of recognition may be customized to the user.

Retraining the system may result when a user has a cold or other conditions which may change the quality of the "trained" utterance. In most cases trainings remain reliable for many months without alteration.

STS

STS executes any of the pages created by Setup and generates activity reports regarding the operation of the software. As each cell is activated by the user, the actions corresponding to that cell such as audio playback and environmental control are activated. In addition, information regarding this activity is stored in a data file. These data files are used to generate reports of system activity and use.

Sound-to-Speech Software Operation

Sound To Speech

Version 1.38

Copyright (c) ARC of the US, 1988-1991

Copyright (c) Genus Microprogramming, Inc. 1988-1990

Copyright (c) Greenleaf Software, Inc. 1984-1989

Press any key to continue

OPERATING SOUND-TO-SPEECH

To activate the Sound-to-Speech program from a DOS prompt, check to see that the computer is logged into the disk drive on which STS is resident. Normally, drive C. If help in change drive assignment is necessary consult your DOS manual for instructions. Once the computer is logged into the correct drive type STS and press ENTER. The copyright screen shown opposite will be displayed, press any key to sequence to the next screen.

Sound-to-Speech Student Information

STUDENT

Add a Student

Delete Student

Edit Student Description

Select Student

Exit

ADD STUDENT MENU

Name: [ ]

Age: [ ]

Diagnosis: [ ]

Person 1 [ ]

Person 2 [ ]

Location: [ ]

Exit and Save

Select Student

BEST COPY AVAILABLE

THE STUDENT MENU

The Student Menu screen that is displayed opposite is the first menu to be presented to the user when the STS software is loaded into the computer.

This menu allows information on new students to be added to the system, student information that is no longer required on the system to be deleted, and existing data to be modified. Additional to the add, delete and modify functions this menu allows a student to be selected for use in all other aspects of the program.

To Add a Student move the cursor, using the arrow keys to that field, and depress ENTER.

Adding a Student

1. To add a new student type a unique name in the Name field, and press ENTER. If a name that is already on record is typed into the Name field, then you will be prompted that the name is already in use and you will not be able to save the name if you attempt to leave this screen.

2. Once a unique name has been enter then the cursor moves to the Age field, type the students age and depress ENTER. If the students age is unknown then this field can be left blank.

3. Enter the students diagnosis, if it is known, or leave the field blank.

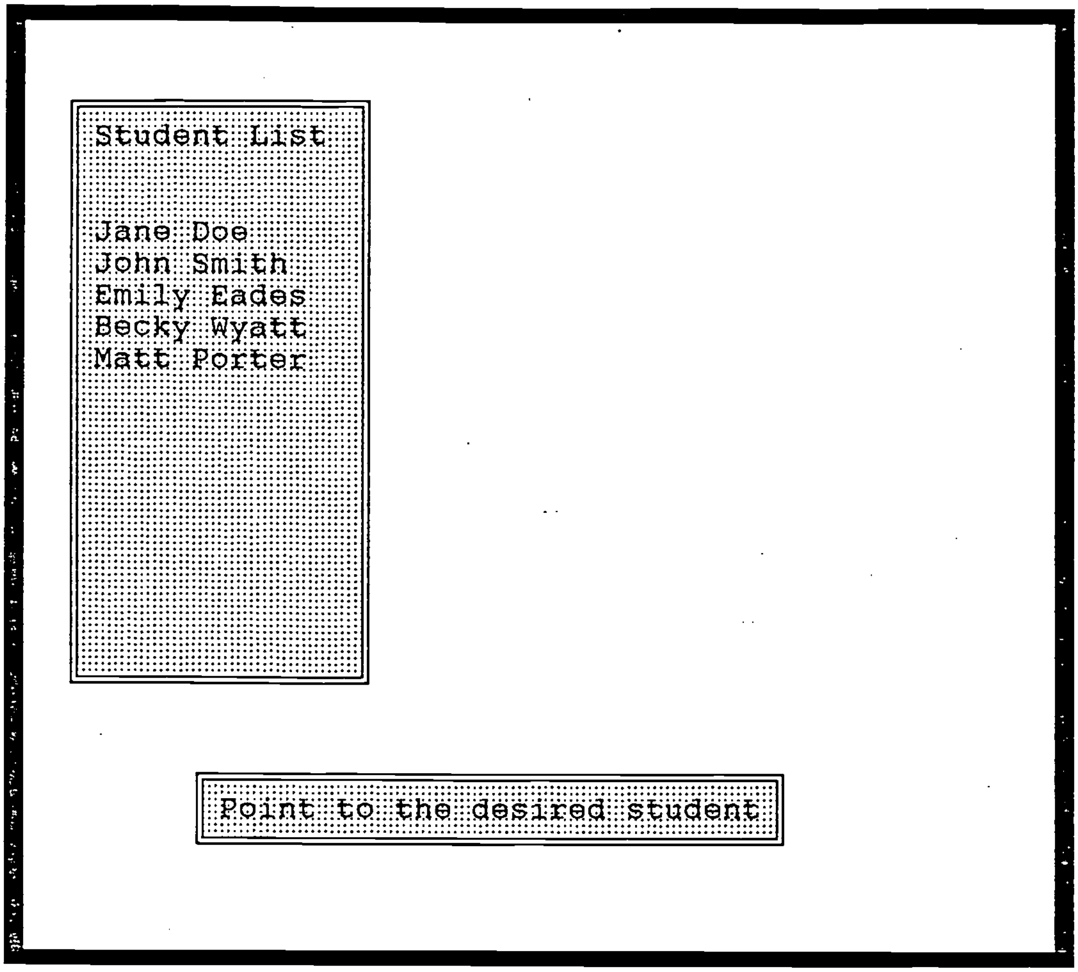
4 & 5. The next two fields on this screen, Person 1 and Person 2 refer to the people assisting the student in using STS, they also can be left blank if the student has a variety of people assisting him/her.

6. Finally, enter the location where the student is going to be using STS. This field as the previous five lines does not have to be filled in.

If changes need to be made to the information that has just been entered, use the arrow keys to move to the line that requires altering and make the correction.

Once all the fields have been filled in correctly then use the Exit and Save field of this screen which allows the information that has just been entered to be saved on the hard disk and the operator to be returned to the Student Menu so that other student information can be added or edited.

If you wish to use the just entered student in other areas of operation then move the cursor to the Select Student field and depress ENTER. This will sequence to the STS Main Menu.


Student List
Jane Doe
John Smith
Emily Eades
Becky Wyatt
Matt Porter
Point to the desired student

Deleting a Student

If student information is to be deleted from the system then move the cursor to the Delete Student field of the Student Menu and press ENTER. The screen marked (B) on the opposite page will be displayed.

Press ENTER to move to the Student List window and move the cursor to the student that is to be deleted, and press ENTER.

A red window will open once you depress enter, and the specified student will be deleted.

Editing a Students Information

From the Student Menu select Edit Student Description to modify student information that has already been recorded.

Once the Edit selection has been made a window displaying the list of students who are saved on the hard disk is opened. Using the arrow keys move the cursor to the student whose information is to be edited and press ENTER.

The Edit Student screen will be displayed. This screen is like the Add Student screen. All entries can be changed by moving the cursor, using the arrow keys, to the line that requires editing and then typing the correction.

Note that the name cannot be changed, the only method that can be used to change a name is to use the Delete Student function and remove the student entirely and then add the student as a completely new entry. This safe guard is necessary to prohibit duplicate names being entered into the computer and thus creating problems when data is to be retrieved.

Once all corrections have been made to the students information press ENTER to save the edited data. This will automatically sequence the software back to the Student Menu.

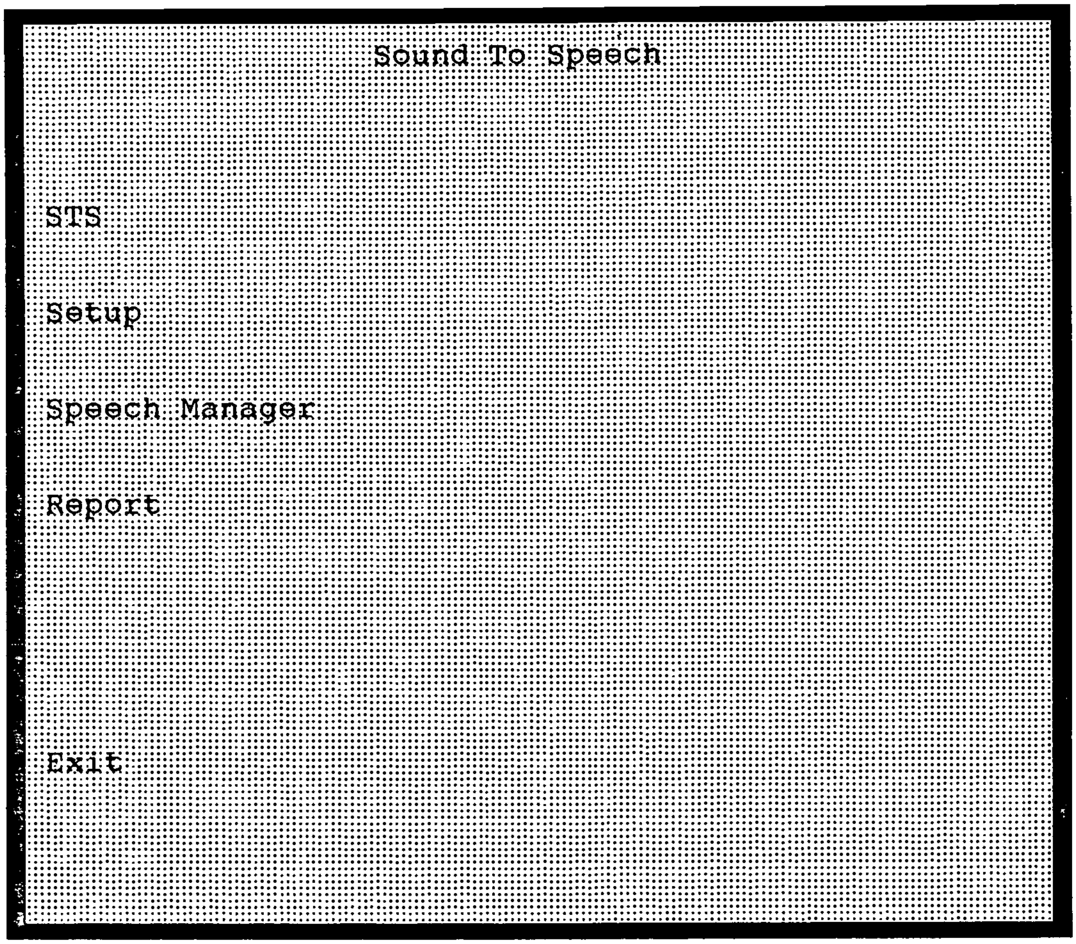
Selecting a Student

To allow STS/Graphics software to function in all other aspects than manipulating student information, then a specific student must be identified and activated. Move the cursor to the Select Student option of the Student menu and depress ENTER to activate a specific student.

A window will open displaying all students who are available for activation, move the cursor to the desired student and press ENTER. This will automatically select the defined student and sequence to the Main Menu.

The Exit option on this menu allows a user to leave the STS/Graphics software and return to DOS. To utilize this function move the cursor to the Exit line of the menu and depress ENTER.

Sound-to-Speech Setup Instructions


Sound To Speech
STS
Setup
Speech Manager
Report
Exit

THE STS MAIN MENU

Once a student has been selected STS/Graphics is able to operate and the Main Menu is available. **The Main Menu cannot be accessed unless a student has been select.** Once the Main Menu has become available then four options are accessible. STS which is the student interactive portion of the software, Setup which allows configuration of 'pages' that will be operated by the STS portion of the program and Reports which provide details of the actions taken during the operation of STS.

The four areas of the main menu need to be operated in the correct sequence. Each section will be discussed to provide insight into its operation, and then the following section will explain the logical succession for the best operation of the STS software.

```
Type the name of the page you wish to construct[          ]
```

```
Scan Method
-----------------
Linear Scan
Direct Select
Row/column Scan
Exit
```

```
Specify the dimension of the matrix (1 to 5) [ _ ]
```

```
Switch Type
-----------------
Voice
Single Switch
```

SETUP

From the Main Menu select Setup and depress ENTER.

The program will sequence to a screen which offers three options; Start a Page, Edit a Page, or Exit. Exit returns you to the Main Menu. Start a Page allows a new page to be created. Once this option has been selected the following instructions are displayed "type the name of the page you wish to construct [ ]. Enter a page name (8 letters or number or combination of both) and press ENTER. The Scan Method screen is displayed once a page name has been entered. There are three access methods offered on this screen. Linear scan, direct select, and row/column scan. See page XX for a detailed description of each of these methods. Move the cursor to the desired field and press ENTER.

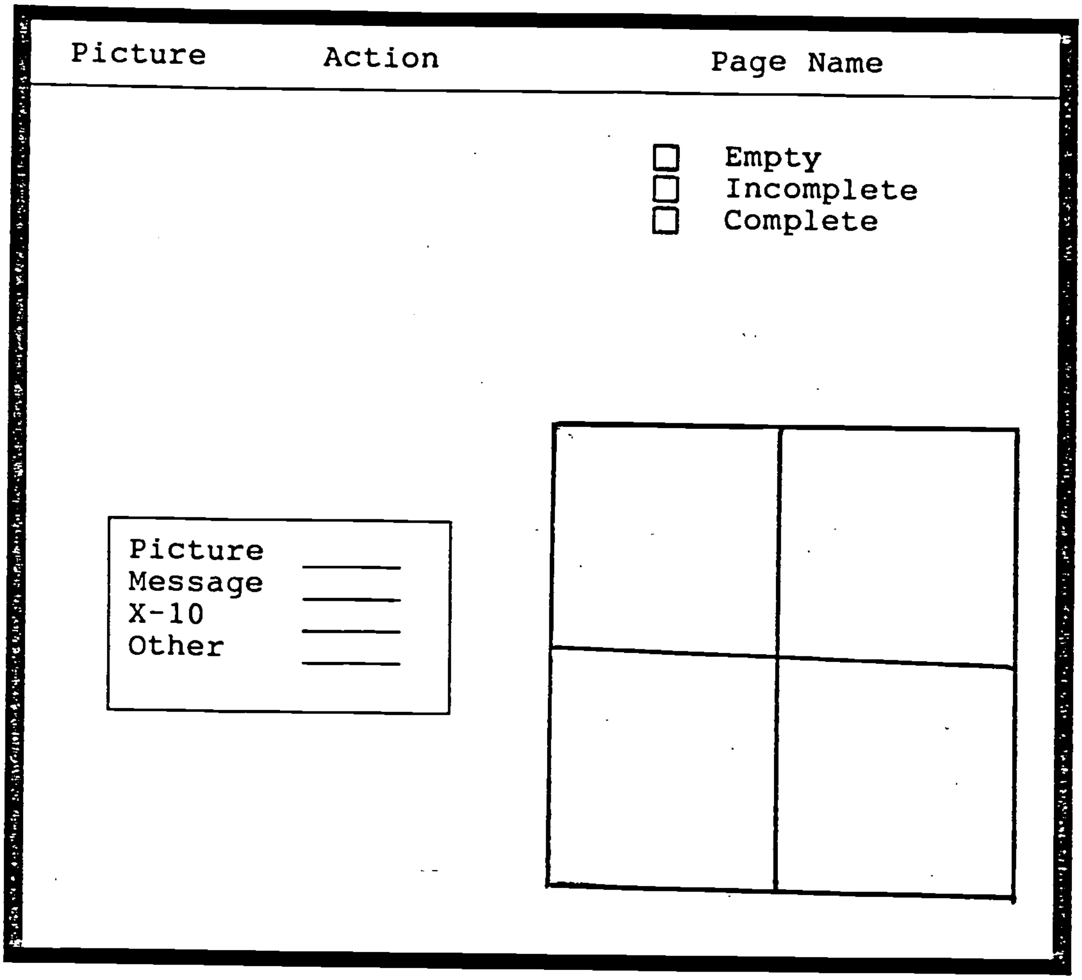
If you do not wish to continue with construction of this page move to the Exit field which will return you to the Main Menu.

Once the scan method has been selected the dimension of the matrix which is to be constructed is requested:

Enter the desired number and press ENTER.

The final screen in the sequence for accessing the Setup mode is used to define the type of switch access that will be used with the specified page. Voice and Single switch are the options, select one and press ENTER.

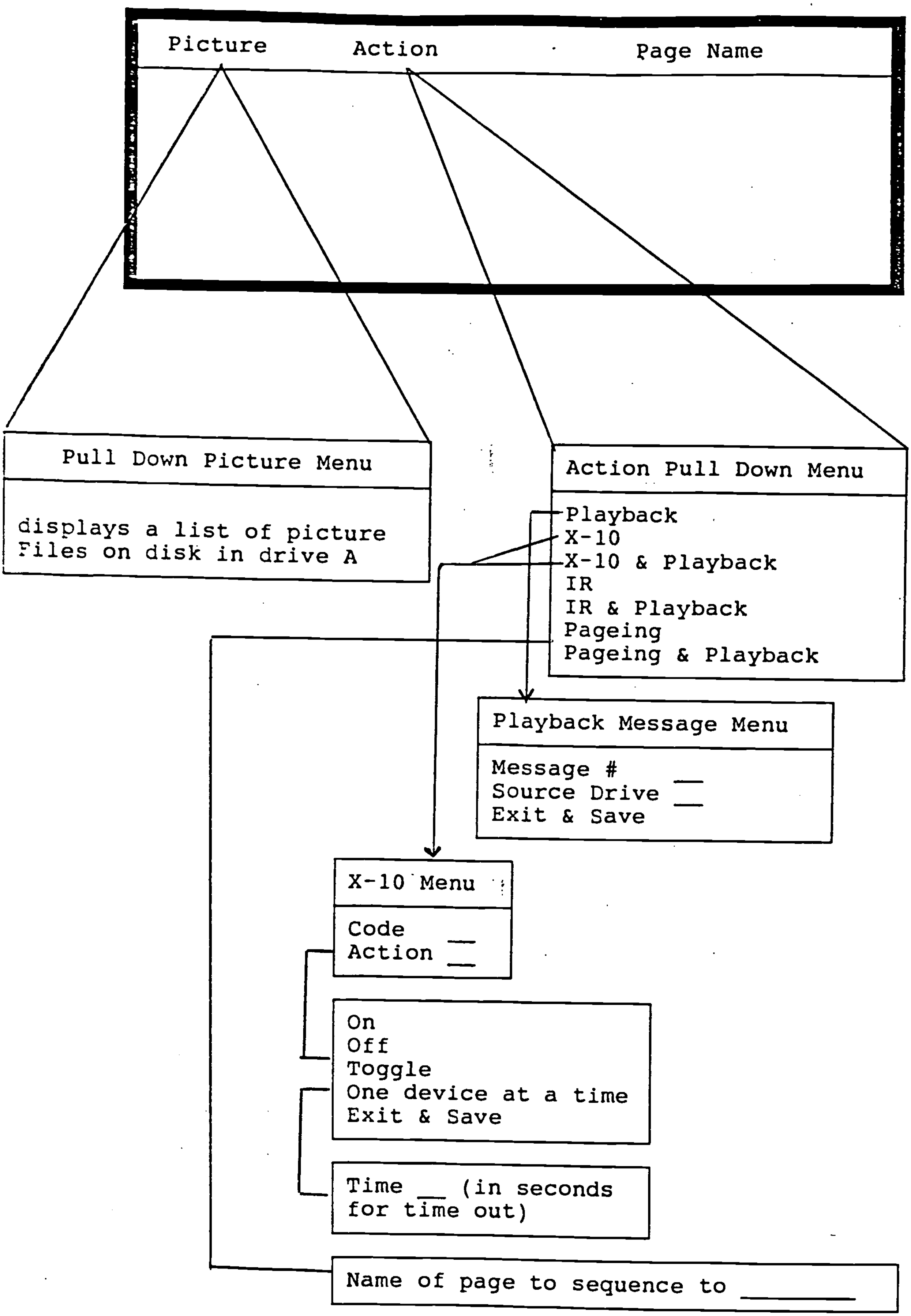
Once all of these selections have been made the Setup screen will be displayed.


Picture
Action
Page Name
Empty
Incomplete
Complete
Picture
Message
X-10
Other

Setup Screen Layout

The setup screen has four major sections. In the bottom right of the screen is an outline of a matrix, this will be laid out with the number of cells which were defined prior to accessing this screen (1, 4, 9, 16, or 25). In the center of each cell in the matrix is a cursor symbol which is colored. In the top right of the screen is a color legend which defines the colors that these cursors can be displayed in. The color is associated with the status of a cell in the matrix. Green indicated that both a picture and an action have been defined for a cell, yellow reflects that only the picture or action have been designated, not both, and red indicates that the cell is empty. In one of the cells of the matrix the cursor will be blinking, this indicates the cell that is currently being configured and all functions both picture and actions that are defined at this time, will relate to that cell.

The box at the bottom left of the screen displays information relative to the cell of the matrix which is currently being configured (i.e., has the blinking cursor). Picture name, message number, environmental control function, and action are all displayed if those functions have been selected.


Picture
Action
Page Name
Pull Down Picture Menu
displays a list of picture files on disk in drive A
Action Pull Down Menu
Playback
X-10
X-10 & Playback
IR
IR & Playback
Pageing
Pageing & Playback
Playback Message Menu
Message #
Source Drive
Exit & Save
X-10 Menu
Code
Action
On
Off
Toggle
One device at a time
Exit & Save
Time __ (in seconds for time out)
Name of page to sequence to ________

Setup Screen Menu Bar

The top left of the screen has a menu line which has two options, Picture and Action, these are activated by depressing P or A from the keyboard, respectively. When Picture is activated the software will look at the floppy disk in disk drive A and display, in a pull-down window, the names of the picture files stored on the diskette. If no diskette is resident in the drive at the time enter is pressed then a prompt to insert a diskette will be displayed. Once the list of picture files is displayed move the cursor to the desired picture and press ENTER. Note that in the lower left box the name of the selected picture file will be displayed.

Action opens a pull-down window that displays a list of action options. This window is illustrated opposite. Each of these options will activate a secondary window and in some cases multiple windows will be opened.

Playback opens a second window as shown opposite. The information required to fill in this window relates to the recorded speech output messages (see section XX for more details). The first field, Message # should be filled in with the number assigned to the desired message, this number is found in the right column of the Message Report. Enter the number and press ENTER. The cursor will automatically sequence to the next field, Source Drive this should be filled in with the drive name where the selected message file is stored. Once both of these field have been filled in correctly move to the Exit & Save field and depress ENTER. Selecting X-10 from the Action Menu opens an additional window as illustrated. The Code field allows the identification of the X-10 wall module number. In this version of STS the house codes, available on X-10 wall modules is configures by the computer as module A. Enter the desired house code number and press ENTER this will move the cursor to the Action field, if the house code that has been entered needs correcting move back to the Code field and make the correction, if the number is correct depress ENTER in the Action field to progress. Once enter has been pressed the following window will open:

This window provides all of the options available for the specified X-10 wall module. On will always turn an appliance on and therefore there should be a second option in the matrix which will provide for the appliance to be turned off, this should be addressed when designing the configuration of each page. Off will only turn appliances off and as described in the previous sentence should have a companion cell in the matrix to turn the appliance on. Toggle will combine these functions in one cell. The first time a cell is accessed by a user, when is defined to toggle, the appliance will turn on. The second time the cell is accessed by the user the appliance is turned off. One device at a time allows only one electronic appliance to be activated at any time. This field activated another window as follows:

Fill in the amount of time that an appliance is required to remain on before automatically turning off. Thus if One device at a time is selected the device turns on, remains on for the given amount of time and then turns off. During the time that this device is on no other device can be activated.

From the Action Menu if X-10 & Playback is selected then the previously described sequences will be sequence consecutively, first the Playback portion and then the X-10 portion.

The IR option from the Action Menu provides access to electronic components that operate with an infrared remote control.

The Paging field of the Action Menu allows a cell in the matrix to be designated as an access cell to another page. If a cell is identified as a Paging cell then when it is accessed by the user the display on the monitor is replaced by a secondary page. On all secondary pages a "return" cell has to be designated, this cell allows the user to return to the original screen.

When the Paging field is selected a window opens will the question Name of Page to page to _______. Enter the name of the secondary page and press ENTER. This will return you to the Setup screen. When the page that is being configured is complete and is being recorded to disk then the user is prompted to designate the "return" cell on the linked page. A window will open during the 'save' sequence which prompts

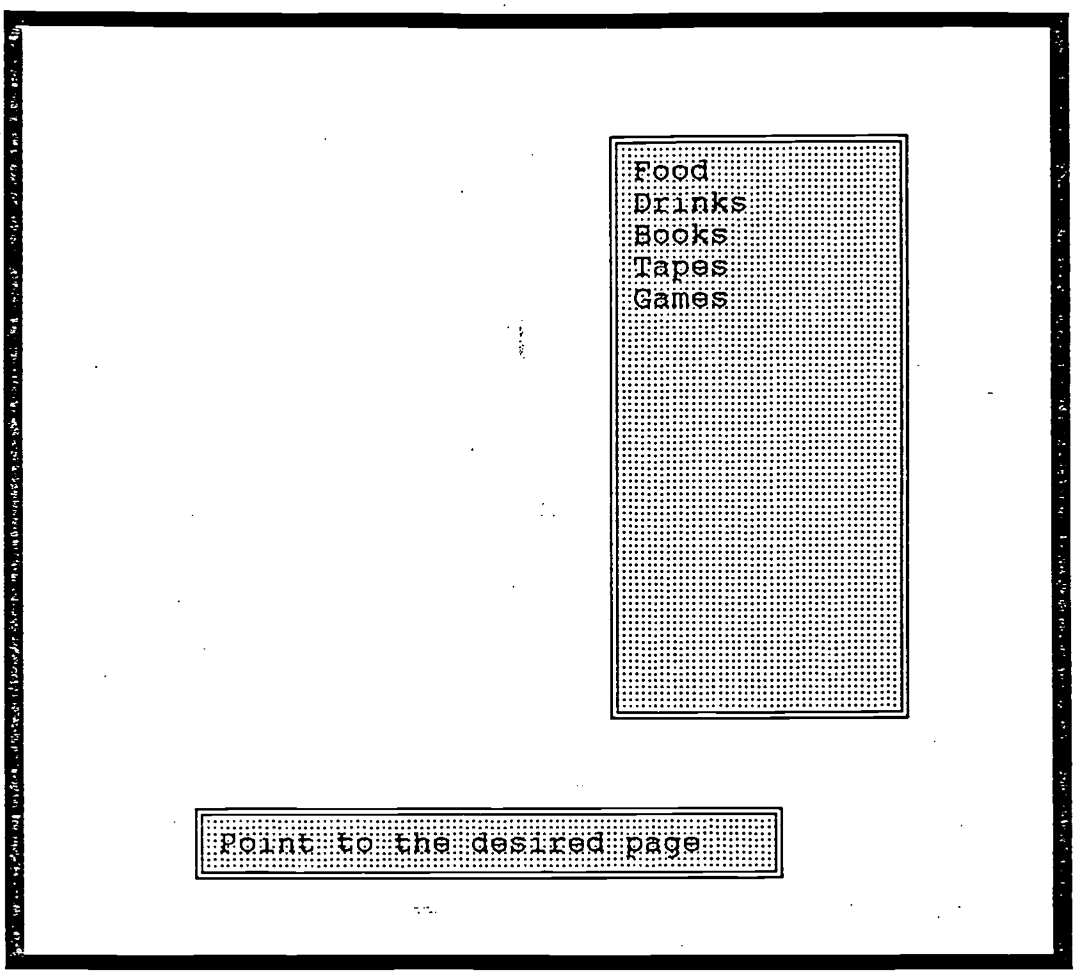
You must choose the return cell for page (name of designated secondary page)

If the secondary page exists on the hard disk then it is loaded into the Setup software and the matrix cursor is located in the desired cell.

If the secondary page has not yet been created then is must be constructed following the Create a Page process and the return cell must be identified. It is not necessary at this time to complete the page, but it must be created. If multiple secondary pages are designated from a page then the previously described process will be repeated until all return cells have been designated.

If Paging & Playback is selected the cell and page are identified followed by the Playback sequence.

This picture and action selection process should be followed until all the cells that are required to be filled are complete. Press ESCAPE to leave the Setup screen. This will automatically save the configuration that has been designated in a file of the name designated at the beginning of the Setup procedure.

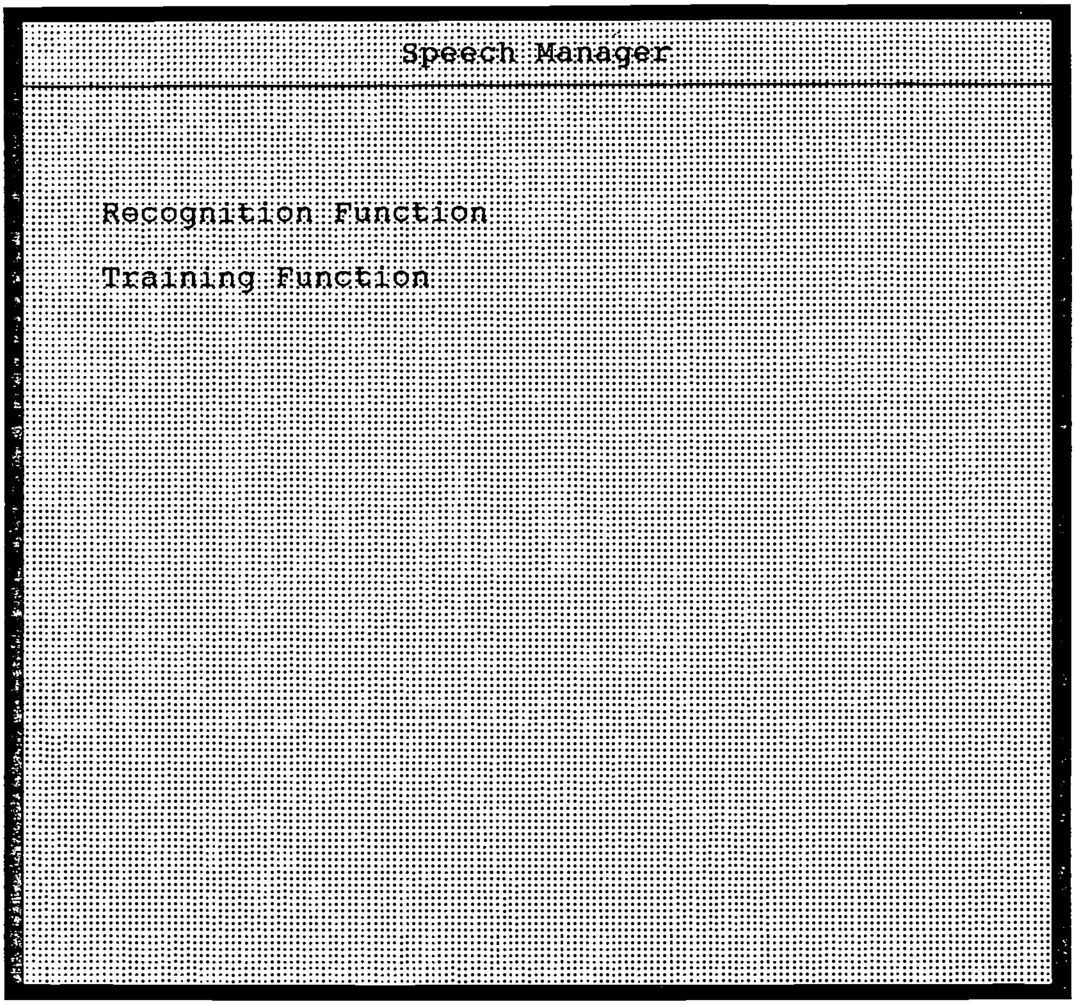


190

Modifying Existing Pages

The edit a page option from this screen allows modifications to be made to an existing page. Move the cursor to this field and depress ENTER to activate the Edit mode. All of the same sequence for editing is followed as is described in creating a new page. For more details refer to the previous section of this document.

Sound to-Speech Speech Manager Instructions


Speech Manager
Recognition Function
Training Function

SPEECH MANAGER

Speech Manager is accessed through the Main Menu. Once it has been selected the Main Menu is replaced by the Speech Manager Selection Menu, as illustrated opposite. There are two options available through Speech Manager, Recognition Function and Training Function. When either of the functions are active they provide information in a windowing format. A window will open and present options or actions and as these options are chosen then second and third window open over the original menu. Follow the instructions to complete all of the functions associated with these windows.

Recognition Function

This function allows the software to be "trained" to recognize a specific students sounds which are used to activate, or act as a switch, for the software when it is being operated through the STS module.

Training Function

The Training Function is used to input and record the messages that the STS software actives when speech output is requested by the student.

Picture	Synonym
Lotion	LO
Music	Mmm
Books	Bu

195

BEST COPY AVAILABLE

Speech Manager Recognition Function

When Recognition Function is selected from the Speech Manager Menu then the screen, illustrated opposite is displayed. There are two column displayed on the screen, A, which lists all of the pictures selected for the student, and B, which is a list of the sounds used by the student in association with each picture.

The Picture list is compiled automatically by the software. Each time a unique picture is selected during Setup to be used in a page file then the name of the new picture is added to the Picture list. If the picture file has already been used then it will not be added to the list a second time. For Direct Selection each picture must have an associated sound. For scanning only one sound needs to be defined.

Four functions can be activated from this screen: Train a Sound, Delete a Training, Help, Modify a Synonym. Each of these functions is activated using the ALT key and a second key simultaneously.

ALT T activated Train a Sound
ALT D activated Delete a Training
ALT H activated the Help screen
ALT M activated Modify a Synonym

Training a Sound

To train a sound move the cursor to the desired picture name and press the ALT and T keys at the same time. A window will open in the center of the screen which requests:

Insert the disk containing "Name of Picture" in drive A and press ENTER"

If you do not wish to continue press the ESC key and control will be returned to the Speech Manager Menu.

To continue with the training sequence place the appropriate disk in disk drive A and press ENTER. The will be a brief pause as the system reads the file from the disk, once this is complete the picture will be displayed on the monitor.

At the top left of the monitor will be a one line window which provides instructions:

"Press Enter to record"

When the student is ready to vocalize press the ENTER key, a beep will sound and the system will wait for a vocalization. The first sound uttered by the student will be recorded and a message will be displayed:

"Save recording Y/N"

If the user feels that the students utterance was appropriate then Yes should be answered to this prompt. If the student produced a sound other than the desired utterance then No should be entered and the sequence repeated. Once Yes has been enter then the complete record process is repeated a second time. Once again if the utterance is not the sound expected to be associated with the displayed picture then the record sequence can be repeated. After two complete cycles through the record process then the system automatically chains to a test procedure for the new recordings. The user is prompted:

"Test the recognition Y/N". If Yes is answered then a beep is emitted and the prompt:

"Say it......" is displayed. Once the student utters a sound the system compares the sound to the two new recordings. If the test sounds is very similar to the recordings then the prompt:

"Good recognitions" is displayed. If the similarity is poor the "Bad recognition" is displayed. Once either of these are displayed then the system presents the option: "Retested the recognition Y/N". Answering No to this prompt will return control to the Speech Manager Menu, answering Yes to the prompt will start the test sequence over again.

This training sequence can be repeated for all of the pictures listed. If a training is omitted then when that specific picture is displayed in a page then it will not be able to be accessed.

Deleting a Training

Once the training sequence has been completed then the sound can be used indefinitely. Over a period of several weeks or months the students production of the specific sound can change, or if the student has a cold then the sound might not be recognized by the system. In the event of this occurring then the recording can be deleted and retrained. If a picture is no longer being used by a student then the sound should be deleted.

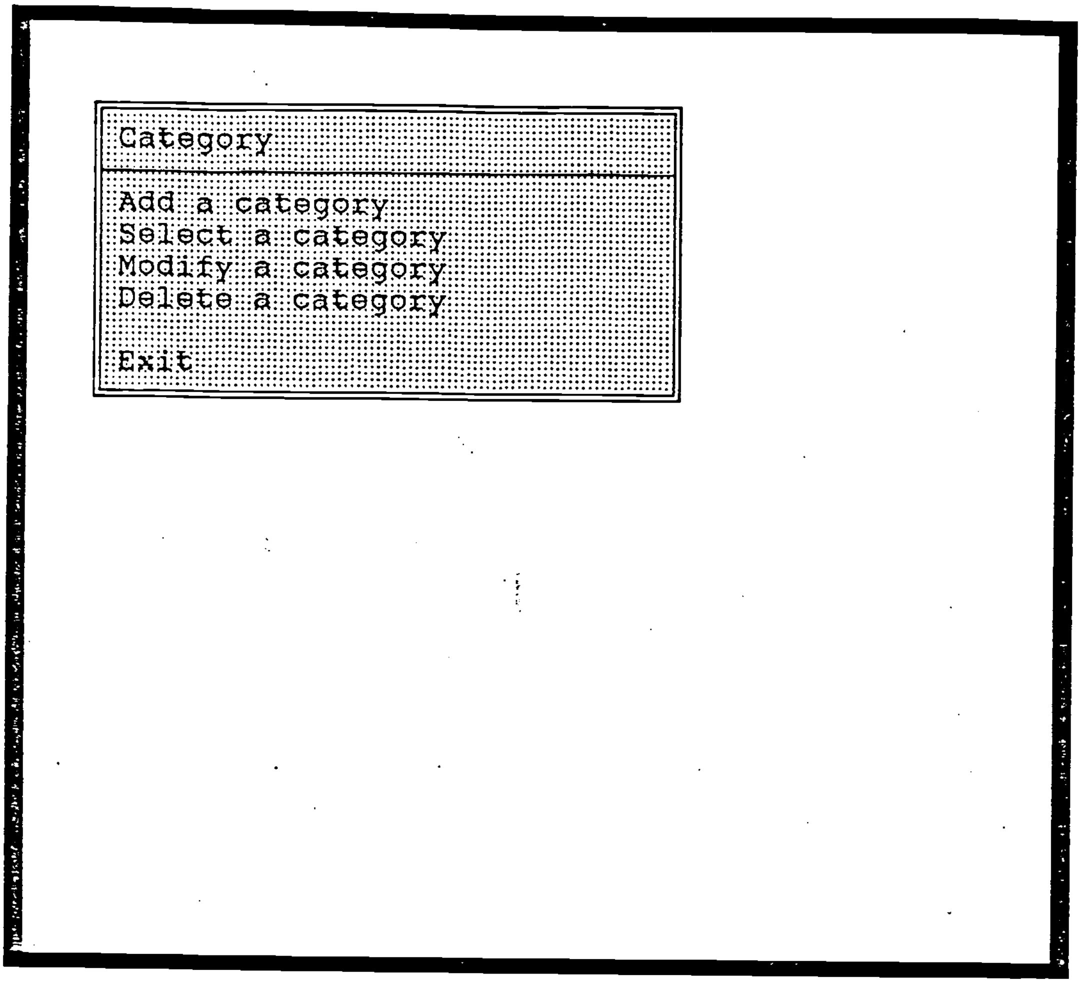
To Delete a Training move the cursor to the name of the picture that is associated with the training. Press the ALT and D keys at the same time. A prompt will be displayed asking "Are you sure you wish to delete this training Y/N". If Yes is answered then the delete process is completed. If No is answered then control is returned to the Speech Manager Menu.

Speech Manager Help

If during operation of the Speech Manager program assistance is needed in how to operate the software then press ATL and H at the same time and instructions will be displayed on the screen. Page up and page down will sequence through the help screens ESC will exit the Help screen.

Modify a Synonym

The synonym list to the right of the screen is used to identify the sounds produced by the student for each picture. To enter or change a synonym press ALT and M at the same time while the cursor is located on the cell that is to be added/changed. Type in the new sound and press ENTER. The new sounds is recorded and will be displayed each time this screen is viewed, until it is deleted or modified.


Category
Add a category
Select a category
Modify a category
Delete a category
Exit

Speech Manager Training Function

The Training Function provides a method of entering messages in both written and audio form. Messages can be divided into categories for easier access and storage and can be printed in this form. Once the Training Function is selected from the Speech Manager Menu then a window opens which presents five options relating to Categories, as illustrated opposite.

It is not essential to divide messages into groups but it is easier to keep them sorted if this function is used. There must be a least one category defined for each user to allow the message recording process to continue.

Add a Category

When Add a Category is selected from the Category window then a second window opens which requests: "Enter Category Name [ ]". Enter a name of up to 8 alpha numeric characters. It is advised that the category name should relate to the types of message which it will contain, e.g., food, games, needs. Once the name has been type then press ENTER which records the name and closes the window.

Select a Category

From the Category window if Select a Category is chosen then a window opens which displays all of the Categories that are presently stored in the students information. Move the cursor to the desired Category name and press ENTER. Another window opens which displays the Message Menu.

Modify and Delete a Category

The Modify and Delete functions from this window are similar in operational procedure and therefore will be described at the same time. Once either is selected from the Category window the list of existing categories is displayed in another window. Move the cursor to the category that is to be modified or deleted and then press ENTER. If the category is to be modified then the name will be displayed in a window which will allow the name to be edited by typing or deleting characters. Once the modifications have be completed then ENTER is pressed and the changes will be recorded and the window closed. If the selected category is to be deleted then the option "Delete Y/N" is presented if Yes is selected then the deletion is completed, if No is chosen then control is returned to the Category window.

Exit

Exit returns control to the Speech Manager Menu. To activate this function move the cursor to the Exit field and press ENTER.

```
Add a Message
Modify a Message
Retrain a Message
Delete a Message
Edit Message Text
Play a Message
Print a Message

Exit
```

Speech Manager Message Window

Once a Category has been selected then the Message Window is accessed. This window provides 8 options which are illustrated on the opposite page and will be described in the following section.

It should be noted that messages are stored in two forms. As text and as an audio message. The options that are provided at times relate to one or other and sometimes both types of messages.

Add a Message

When Add a Message is selected from this window then a secondary window open with the instruction, "Type the message" Enter the text of the message which will be recorded for playback during STS operation. Once the text is entered press ENTER the window is replace with another window which instructs, "Press ENTER when ready to record". Check to see that the person who is going to record the message is correctly positioned in front of the microphone and press ENTER. The person recording the message should begin talking as soon as ENTER is pressed. The software "listens" for the silence after the message and as soon as the speech halts the system ends it recording process and displays a message saying ".........Got it". The dots appear as the speech occurs. The window now displays a message "Press ENTER for playback", this initiates playback of the message which has just been recorded and gives the user the opportunity to hear what the message sounds like. Once the message has been played back then the option "Re-record the message Y/N" is displayed. Answering Yes at this prompt will return the user to the beginning of the record sequence. Answering No to this question returns control to the Message Menu.

Modify a Message

When Modify a Message is selected from the Message window another window is opened which displays a list of the messages that have already been recorded. To the right of the text a number is displayed. This is a number which is automatically assigned to each message by the software and is used during the page setup procedure to define which message will be played back. If more messages are stored than can be displayed in the window then use the down arrow to scroll to the lower messages. Once the desired message is identified by the cursor press ENTER. A window will open with the message displayed in it. Use the arrow keys to move to the part of the message that is to be altered and type in the correction. Once the text is corrected then the procedure is the same as Adding a Message.

Retrain a Message

Selecting Retrain a Message from the Message window activates the list of messages, select the message that is to be retrained from the message list by locating the cursor on the text and pressing ENTER. The software will sequence to the record function that is explained in the Add a Message section above.

201

Delete a Message

If Delete a Message is selected from the Message window then the list of messages is displayed. Move the cursor to the message that is to be deleted and press ENTER. The prompt "Are you sure you wish to delete this message Y/N" will be displayed, if Yes is selected then the message is deleted and control is returned to the Message window. If No is selected then no action takes place and control is returned to the Message window.

Edit Message Text

When Edit Message Text is selected from options in the Message window then the list of messages is displayed. Select the message that is to be altered by locating the cursor on it and pressing ENTER. The selected message will be displayed in a window and alterations can be made to the text by retyping the message, using the delete or arrow keys or a combination of all of these keys. Once the message is modified press ENTER to record the altered text. Control will be returned to the Message window.

Play a Message

As with Delete and Edit a message once Play a Message is select from the Message window options then the message list is displayed, select the message from the list that is to be played and press ENTER. The audio message associated with the selected text will be played and control will be returned to the Message window.

Print Messages

When Print Messages is selected from the Message window then a second window opens asking if the printer is ready, check to see that the printer is connected to the correct port on your computer and press Y. A list of all message stored for the selected student will be printed by category. See the illustration on the opposite page.

Exit

Exit returns the software to the Speech Manager Menu.

Sound-to-Speech Reports & Data Analysis

The Arc
SOUND-TO-SPEECH

STUDENT SUMMARY

A
Date/Time: Wed Jan 29 09:14:37 1992
Name: Maggie Sauer
Age: -
Diagnosis: -
Assistant 1: -
Assistant 2: -
Location: -

DISPLAY CONFIGURATION

B
Access Method: Direct Select
Page: DIRECT
Matrix Size: 3 x 3
Initial Recognition Distance: 50
Matrix Color: Black
Scan Color: Green

CELL INFORMATION

C

Cell 1:
Synonym: Message: 0

Cell 2: LIPGLOSS
Synonym: LIPGLOSS Message: 3

Cell 3:
Synonym: Message: 0

Cell 4: REDPOL
Synonym: REDPOL Message: 1

Cell 5:
Synonym: Message: 0

Cell 6: LOTION
Synonym: LOTION Message: 2

Cell 7:
Synonym: STIC Message: 0

Cell 8: STICKER
Synonym: STIC Message: 4

Cell 9:
Synonym: STIC Message: 0

DATA COLLECTION AND ANALYSIS

A chronological report is generated of all system activity is generated whenever the system is used. Subsidiary reports are generated from the chronological report detailing specific activities such as environmental control and audio playback. The subsidiary reports include an analysis of these activities.

Student Summary

The Student Summary is divided into three main sections; identifying information, display configuration and cell information.

The identifying information section (a) is provided to label training dates, participants and location. Since user performance may vary with each of these variables, this information is important complete each time the system is used particularly in the early stages of system training.

The display configuration section (b) provides a summary of how the system was setup for the user. It describes the access method, matrix size, recognition level, matrix color and scan box color. This information is helpful in comparing user performance as these variables are changed as well as referencing a change in performance if the variables are unintentionally changed during the training process.

The cell information section (c) provides a written description of the contents of each area or cell on the display. This includes the description of the picture in the cell as well as the message which will be spoken by the computer when the cell is activated by the user.

The Arc
SOUND-TO-SPEECH

A
Date/Time: Wed Jan 29 09:14:37 1992
Name: Maggie Sauer Age: -
Diagnosis: - Assistant 1: - Assistant 2: -
Location:-

B
Page: DIRECT
Access Method: Direct Select Matrix Size: 3 x 3
Initial Recognition Distance: 50
Matrix Color: Black Scan Color: Green Beep: On

C

Event	Box #	Time	Cell Name	Device	Function
	2	09:14:37	LIPGLOSS	Playback	I want to put on lipgloss
	8	09:14:41	STICKER	Playback	Stickers are sexy
	4	09:14:46	REDPOL	Playback	I love bright red nails
	6	09:14:50	LOTION	Playback	My hands are dry, I need lotion
	4	09:14:54	REDPOL	Playback	I love bright red nails
	2	09:15:11	LIPGLOSS	Playback	I want to put on lipgloss
	8	09:15:14	STICKER	Playback	Stickers are sexy
	4	09:15:25	REDPOL	Playback	I love bright red nails
	6	09:15:30	LOTION	Playback	My hands are dry, I need lotion
	8	09:15:33	STICKER	Playback	Stickers are sexy
	4	09:15:38	REDPOL	Playback	I love bright red nails
	2	09:15:43	LIPGLOSS	Playback	I want to put on lipgloss

CHRONOLOGICAL REPORT

The chronological report is divided into three main sections; identifying information, display configuration and event.

The identifying information (a) and display configuration (b) information is identical to the information provided on the Student Summary page. Duplication of the information on the chronological report will assist staff in matching the student summaries and chronological reports for future reference.

The event section (c) enables the facilitator to track item selection and system use across time. This is information is useful in determining:

a. Preferences
b. Functionality and use of items on the display
c. Time required by the user to make selections
d. Errors produced during training sessions.

This information will assist the facilitator in making changes in system options, the actual items available to the user on a display, or the number of items provided for the user.

Additional information regarding data collection and analysis has been provided in Chapter 8 - Case Examples.

Sound-to-Speech STS Operation Instructions

STS

When Student Information, Messages and Recognition Trainings, and Page Setup have been completed the software is ready for operation. Select STS from the Main Menu to run the interactive portion of this software.

Once STS has been selected then a list of pages is displayed. Select the page that is to be operated, from the list by positioning the cursor on the desired page name and pressing ENTER.

The screen will clear and then the page will be displayed. Depending upon the configuration in Setup the page will function differently. If access was designated as through scanning then once the page is displayed the scan box will start to move through the matrix in the predefined manner, (linear, or row/column). The scan box can be halted by the student producing an utterance.

If access was through Direct Selection then the page will be displayed and be immediately in a active mode. It will wait for sounds and once a sound is produced then the system will try to match the sound to a recorded utterance. If the system finds a similar sound then it will execute the appropriate functions that were defined to activate with the match of that sound.

Function Keys

There are nine keys that can change how the STS software reacted during operation:

F1 Increases the speed at which the scan box moves through the matrix.

F2 Decreases the speed at which the scan box moves through the matrix.

It should be noted that each time either key is pressed the scan box must move one cell before the key stroke is acted on, e.g., if F1 is pressed twice then the scan box will move two cells before the box moves faster.

F3 Acts as a toggle and turns the beep, which is associated with the scan box movement on or off.

F4 Changes the matrix color. Each time F4 is pressed the matrix color changes through green, blue, red, yellow, and black.

F5 Changes the color of the scan box through the same colors as the matrix. The scan box colors sequence independently from the matrix color changes.

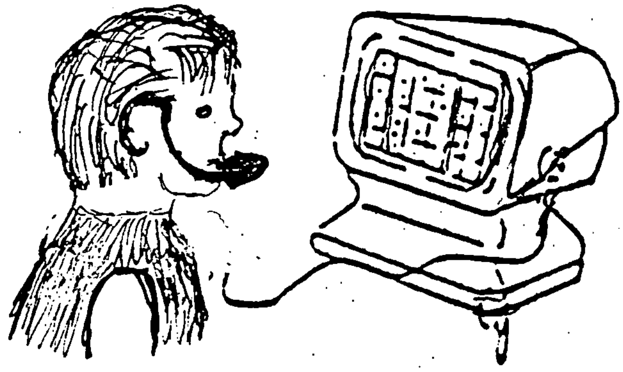
F6 F6 controls the start position of the scan box. It can either always return to the top left cell in the matrix when the scan is halted, or continue on to the next cell in the matrix after the scan is halted.

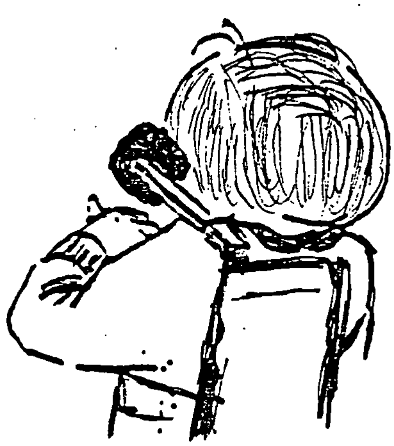
F7 F7 provides a method of triggering message output from the keyboard for a specific cell. This function is used during the early stages of training to provide feedback to the student even when their vocalizations are not accurate.

\+ The + key can be pressed repeatedly to increase the recognition of the system for utterances. If the recognition is widened then a greater variance in the utterances produced by a student will be match to the trained sounds in the system.

\- The - key decreases the recognition rate of the system and makes it less able to match utterances to the trained sounds. As a student becomes more able to produce sounds consistently then the recognition can be decreased. This allows more sounds to be used with the system without them being miss matched.

Section 3
Initial Considerations





INITIAL CONSIDERATIONS

Setting up the STS System

There are some basic considerations which can enhance the operation of the system by the user. These include monitor height, microphone placement, and switch placement.

Monitor height. The monitor height should be adjusted to a comfortable level for the user. The user should be able to scan the display without unnecessary vertical or horizontal head movement.

Microphone placement. As discussed in Chapter 1, there are a variety of microphones which can be used by the STS system. Individuals assisting the user with system setup and microphone selection should take care to read manufacturer instructions regarding placement. These recommendations should be closely followed to ensure effective system operation. If a microphone is improperly placed, it can dramatically effect the ability of the system to recognize the user's speech sounds.

Switch placement. Appropriate switch selection and placement are critical to the user controlling the STS system in this way. For many individuals with physical limitations which require them to use a switch, careful switch placement is the critical difference between independence and dependence. Evaluation of the individual's fine and gross motor skills is necessary to effectively determine placement. Information can be obtained from an occupational therapist assisting the user and/or caregiver.

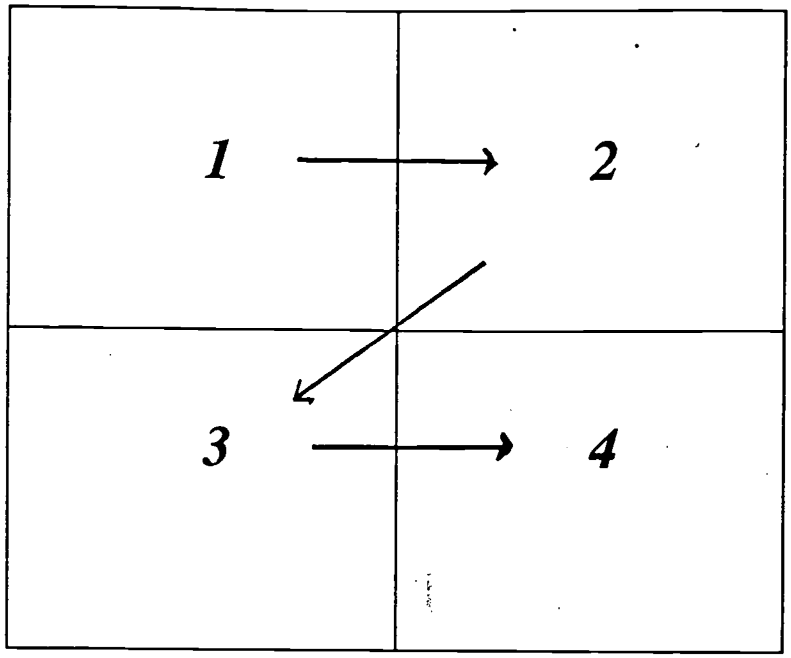
CUSTOMIZING DEVICE OPERATION FOR THE USER

The STS system is designed to adapt to the needs of the user. These features were discussed in Chapter 1 but will be discussed here with regard to strategies for implementation.

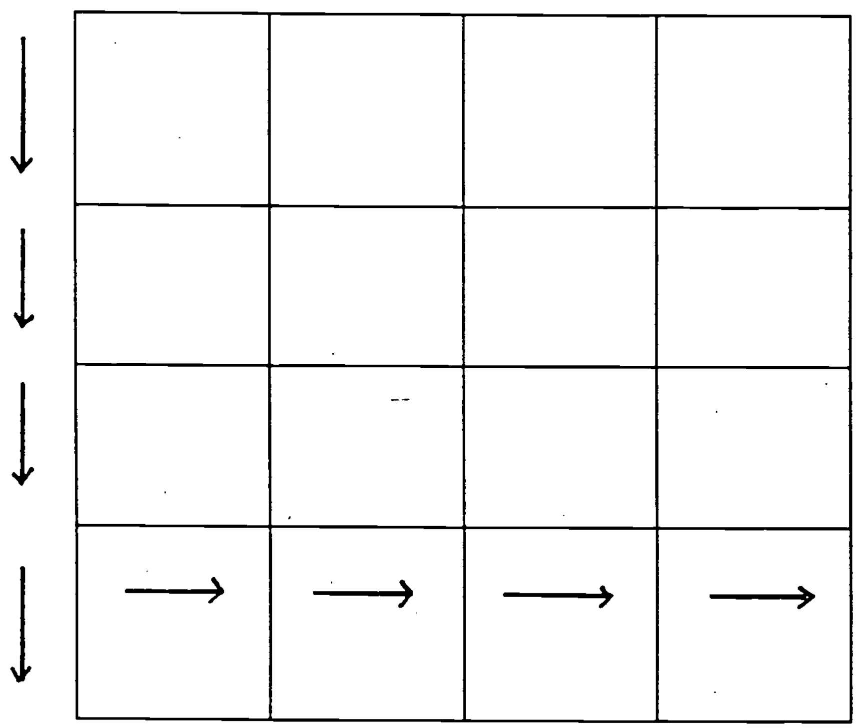
Choices/Selection of Photos for the Display

In order to organize the photos which will be used for constructing the user's communication displays, it is recommended that caregivers generate a list of objects/activities/people the user communicates about. After this list has been constructed, it will be easier for the caregiver and/or user to group similar items together. These items can then be grouped together on one display to create a topic board or categorically related group of items or messages. Most importantly, however, is the selection of items which are meaningful to the user (Please refer to the appendix for additional information).

Research has shown that for persons with cognitive impairment, the use of items from their natural environment as well as training which occurs in these settings is most effectively generalized. The effectiveness of the STS system comes from the capability to implement photographic quality images of items/people in the user's natural environment as well as the user's own speech sounds.



LINEAR SCAN - The scan box moves from box to box sequentially



ROW/COLUMN SCAN - The scan box moves between rows until it is halted by the user on a specific row. Once the row has been selected the scan box then moves across the row cell by cell until the user activates a switch that causes the scan box to stop

Customizing the Display for the User

There are several options available which can enhance user operation. These include scan speed, recognition level, matrix/scan box color, and beep/signal. Each of these features may be adjusted from the keyboard while the STS system is being operated by the user. This permits the customization or "fine tuning" of the system as needed. The following keys on the keyboard can be used to change the systems operation:

Scan speed The rate at which the scanning box moves across the display in either the linear or row/column scanning mode is controlled by:

F1 which increases the scan speed
F2 which decreases the scan speed

Examples of each scan type have been illustrated opposite.

Recognition Level This refers to how closely the user's speech sounds must be produced, in comparison to the trained utterances, to be recognized by the system.

+ Key expands the recognition level
- Key narrows the recognition level

Matrix Color Matrix colors include yellow, red, blue, black and green and may be changed by repeatedly activating the F4 key .

F4 toggles the matrix colors

Scan Box Color Scan box colors include yellow, red, blue, black and green and may be changed by repeatedly activating the F5 key.

F5 toggles the scan box colors

Beep/Signal The beep indicates scanning box movement on the display. It can assist the user in more effectively attending to the display.

F3 toggles the beep on/off during the scanning mode

The user should be exposed to each of these features and permitted to use them in a variety of combinations. Information should be recorded by support personnel or family members to determine which settings most effectively enhance user performance.

WORKSHEET

Name:

Activity/Item	Speech Sound	Message	Photograph

Additionally, the user's sound for the item or activity should be recorded with the photographic image of the item on the display. The worksheet provided opposite permits the individual assisting the user in setting up the system to organize these speech sounds, images and accompanying messages.

Examples of system operation for scanning as well as direct selection using voice control are contained in the Chapter 8-Case Examples. Examples of the system settings and user performance are provided.

Section 4
Training Considerations

WORKSHEET

Name:

Activity/Item	Speech Sound	Message	Photograph

TRAINING CONSIDERATIONS

In Chapter 3, some of the initial considerations for system use and customization were addressed. This chapter will discuss some of the training considerations related to use of the direct selection or single switch scanning mode.

Direct Selection-Voice Control

In order for a user to effectively control the STS system through direct selection or voice control, they should be able to produce a variety of sounds. These sounds may be unintelligible, yet, must be able to be repeated consistently. This consistency is necessary for the STS system to recognize the user.

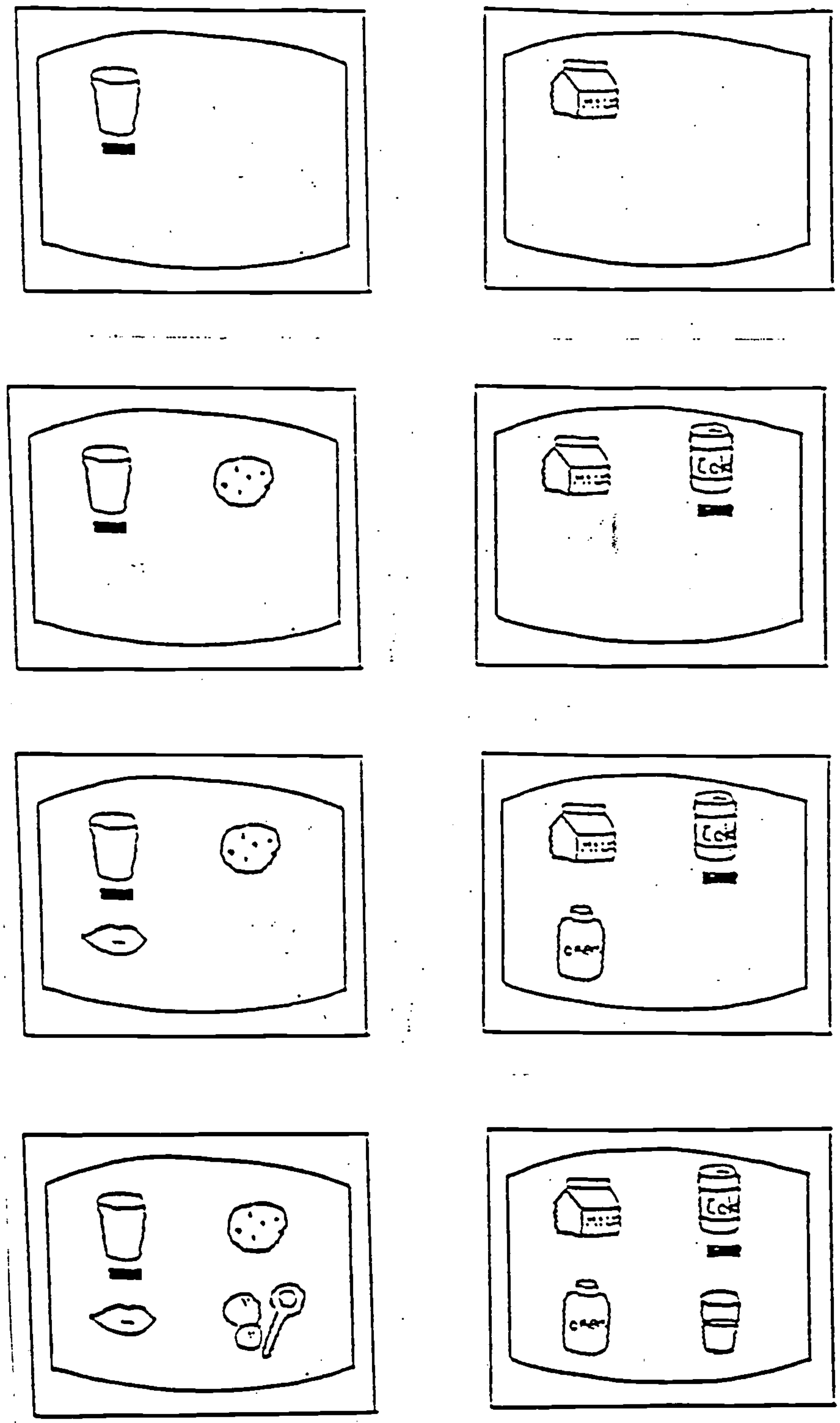
In order to begin making these determinations, interviews should be conducted with parents, caregivers, and other support staff regarding the user's speech and repertoire of speech sounds. Many times, these individuals can identify speech sounds which may be used in reference to specific activities, events, objects or people. These sounds become the initial training point for the STS system. Particularly for individuals with cognitive impairment, structuring these training activities in this way can be critical for the following reasons:

a. Enables the user to implement a speech sound they presently use as a meaningful message which will be provided by the computer.

b. Utilizing their own speech sound rather than teaching sounds, the user simply has to learn that they are controlling the system with their voice rather than a sound which has been selected by others.

c. Cognitively this task is easier because the user has already developed an association between the speech sound and the activity or event.

In Chapter 3, creating a picture library using a vocabulary list was recommended to customize the system to the user's needs. Creating a message library which describes the user's speech sounds and their meaning can be used to train the user with the appropriate communication display. Furthermore, these speech sounds can be used to train the system to recognize them and thereby "translate" these sounds into messages spoken aloud by the computer. In order to document these speech sounds and their associated referents, the table below has been provided as an example of how this may be organized. In this way, a list may be developed of the user's speech sounds, their referent activities, objects or people and the intended message. The lists will enable personal care assistants, parents, and teachers to organize this information and place it in the STS system.



Training for the STS System should begin with very simple activities to give the user confidence and success in controlling the system. Additionally, it is best to begin with activities the user will successfully complete and gradually increase the complexity of the activity. For example, the user may begin with one item on the display. As the user gains control over the system with single items, additional items may be added to the display.

It may be necessary to adjust the user's recognition levels for the system as items are added. Following the training procedures described in Chapter 2, the system should be trained to recognize the user's speech sounds to insure reliable system performance. It is recommended that the user "test" each item on the display to insure maximum response by the system. The system should be "retrained" for any item which the system fails to adequately recognize.

Single switch/Voice

Initial Considerations Seating/positioning, switch selection and switch selection are critical ingredients to the effective use and implementation of the STS system. The appropriate adjustments and combination of these three factors provide the foundation for optimal system use. If additional assistance is required to address any of these areas, a physical, occupational, and or speech/language pathologist may be consulted.

During the initial stages of training it may be necessary to provide the user with opportunties to practice using their switch and improve their control in operating it. A variety of opportunities may be provided outside the STS system to assist them in developing these skills. Examples of these activities may include controlling small household appliances, simple computer games, battery operated toys etc. For the user who does not have a range of speech sounds but prefer to use their voice to activate the system, voice activated switches are available and may be used in the manner described above. These voice activated switches are available from a variety of manufacturers and range in price and effectiveness. Additional information has been placed in the appendix which describes switches, switch mounting hardware, and computer software programs which may be used for some of these activities.

When using a switch or the user's voice to control the scanning display on the STS system, a hierarchy of tasks should be provided. It is not necessary to follow these steps in order, however, they are provided as an outline of sequenced training steps which may be helpful to begin to gradually develop skills to enable them to effectively control the STS system. It should be noted that the suggested training strategies are not unique to the STS System. These strategies may be applied more generally to training controlled switch use and scanning. They are provided here to assist parents, caregivers and teachers in the use and training of the STS system specifically.

Training for the STS System should begin with very simple activities to give the user confidence and success in controlling the system. Additionally, it is best to begin with activities the user will successfully complete and gradually increase the complexity of the activity. For example, the user may begin with one item on the display. By activating their switch, the STS system would then speak the accompanying message and/or turn on the accompanying appliance. This activity will give the user the opportunity to learn that their switch activation has caused something to happen on the STS system. When the user understands this relationship between the switch and the item on the STS system, it is possible to move on to a simple scanning activity using one item on the display.

At this time, the system should be further customized for the user. The options for customization of the STS system were discussed in Chapter 3. They include:

scan speed
recognition level
matrix color
scan box color
beep/signal

A variety of settings should be used to determine the user's optimal performance for each setting. With regard to scan speed, the user may begin with a speed setting of 5 and adjust the setting depending on the performance of the user. Each of the settings should be used in this way and adjusted as necessary. It should be noted that user's performance may vary across time due to fatigue, attending behavior etc. These settings should be adjusted for the user's performance accordingly.

After adjusting settings for the user, a simple scanning task may be introduced. A single item in the 2x2 matrix may be presented. The user may be given opportunties to practice using the single item in a 2x2 matrix in order to develop the visual scanning skills and timing necessary to activate the switch when the scanning box is located on the desired item. Initially, this should be done in the linear scanning mode. As the user's proficiency with this task improves, additional items may be added to the display. Adding items to the user's display is discussed in detail in Chapter 2 - STS Desciption and Operation.

A user's speed during the scanning process may be enhanced utilizing the row/column method of scanning. This is particularly useful for larger display sizes such as 3x3, 4x4 and 5x5. However, it should be noted that the row/column scanning method requires more skill by the user. It requires the user to locate the desired item on the display and plan for the selection. This is done by activating the switch in the row of the desired item and a second switch activation as the scanning box advances to the item within the row. This requires additional cognitive planning as well as switch timing skills. The user should be given opportunities to practice these skills before they are asked to locate desired items for communication and/or environmental control purposes.

Choices

The user should be provided with choices which are meaningful and motivating to them in their environment. This includes home, school and community activities. It is important to include choices or opportunities for communication which the user wants to communicate about. Choices should be provided to the user on the STS system which provide these opportunities.

Display Configuration

The number of choices offered to the user will depend on their abilities. This can vary particularly with regard to how the individual is controlling the system. Most importantly, it is important to begin training the system with one or two items. As the user's proficiency with the system improves, the number of items should be increased accordingly.

Additionally, as discussed previously, it may be necessary to adjust the scan speed, matrix color, etc. to best accommodate the needs of the user.

Additional information has been placed in the appendix of this training manual which further describes the construction of choice-making opportunities, training strategies and constructing topic boards. Topic boards are categorically specific communication boards which resemble the choice-boards constructed for the STS system.

Section 5
Case Examples

FIELD TESTING THE SOUND-TO-SPEECH SYSTEM

Subjects were referred by teachers, therapists and special education directors in the cooperating schools. For the evaluation phase of this project 24 subjects (12 male and 12 female) were evaluated ranging in age from 4-21 years. Diagnosis for individuals evaluated, included cerebral palsy, autism, encephalitis, Down's syndrome, and renal dysfunction. Subjects varied in range of mental retardation from mild to profound. Of the 24 subjects evaluated, 14 were ambulatory and 10 were non-ambulatory. Criteria for selection to the Sound-to-Speech Project included:

- o adequate vision and hearing
- o adequate positioning and seating
- o understanding of picture/object association
- o understand cause/effect relationships
- o possess rudimentary scanning skills (if appropriate)
- o speech is not functional as primary mode of communication

Six subjects were selected for participation. Final subject information is described below:

Female: 4
Male: 2
Age Range: 7-21
Diagnosis: cerebral palsy, encephalitis, mental retardation - mild to profound
Ambulatory: 4 (Two walk with the assistance of a walker)
Non-ambulatory: 2

Case examples of system use were generated from these field testing activities and summarized for a user who controlled the system utilizing direct selection and a user who controlled the system utilizing a switch with scanning.

Single Switch Operation - Demmie

Diagnosis - Moderate to severe mental retardation

Age - 21

Etiology - Origin of mental retardation unknown, restricted movement for walking, nonspeaking with the exception of a few speech sounds, speech was preferred method of communication regardless of intelligibility to others.

History - Demmie was a young woman with moderate to severe mental retardation. She had been introduced to a variety of communication methods which included sign and communication boards. Each of these methods had been relatively unsuccessful because of Demmie's preference to use her speech over all other methods of communication. She attended school and was limited in her ability to reliably communicate with her teachers, peers and family members. Demmie relied on their ability to "translate" her communication attempts across each of these environments.

Recommendations for System Use - Although Demmie could not control the Sound-To-Speech system using individual speech sounds to represent each of her messages, she was able to use her voice as a switch to control the system through scanning. She was motivated and interested in this method of communication and using the system to assist her in accomplishing this.

A list of Demmie's preferred activities and communication messages were generated using the worksheets found in the appendix to this manual. Displays were prepared using these worksheets and pairing appropriate photographs for each of the items on the displays. Beginning with one item on the 1x1 display, Demmie gained more confidence in her control and operation of the system. As her skills improved additional items were added to her displays which consisted of beauty aids, leisure time activities and drawing or coloring activities.

System Use - It was necessary to vary the scanning speed of the scanning box as Demmie's skills improved. Additionally, two different colors were used to highlight the matrix color and the scanning box. This assisted Demmie in visually identifying the scanning box on the display as well as division on the display between her choices or pictures.

The data collection tools available on the STS system assisted staff in comparing Demmie's performance as these variables were changed. Furthermore, this assisted staff in making changes which improved her overall performance and control with the system. Staff were also able to discern which items were really functional and useful to Demmie on the displays. Thus, items which were not used or non-preferred could be replaced with more functional and motivating messages and pictures.

Voice Control - Misty

Diagnosis - mild to moderate mental retardation

Age - 7

Etiology - Unknown, nonspeaking, demonstrated the ability to produce a variety of speech sounds which she used to reference specific activities, objects and people in her environment.

History - Misty was diagnosed as having mild to moderate mental retardation. Like Demmie, a variety of communication systems had been tried including sign and communication boards. Misty preferred to use her own voice to communicate and used this strategy most often with her parents at home. At school, Misty was described as "being quiet" and "not talking very much".

Recommendations for System Use - Misty and her mother were asked to bring a variety of her favorite activities and toys with her to test the system. Her mother reported different speech sounds used by Misty with the six items provided for use with the system. This was confirmed by Misty's production of different speech sounds each time the items were used during the introduction. Photographs of each of the items were scanned into the system and presented on the system display. The system was trained to recognize Misty's speech sounds for each of the six items. Misty demonstrated the ability to produce each sound as the picture image of each item was presented.

Displays were constructed for Misty using the information provided by her mother and school staff regarding Misty's preferred activities and people.

System Use - Misty began by using the system one item at a time and gaining confidence in using the system. As her skill and control improved items were gradually added to her displays one by one.

The data collection tool for the STS system enabled staff to keep track of her performance with regard to errors, items used and not used and performance variance with different instructors.

Overall, parents and school staff reported increased vocalization, interaction and improved speech intelligibility.

Appendices

Training Strategies

TOPIC BOARDS- ONLY THE BEGINNING

Maggie Sauer M.S., CCC-Spl
John C. Costello, M.S., CCC-Spl

The following was modified based on information available through the Communication Aids and Systems Clinic, Madison, Wi.

Miniboards

1. Miniboards are small topic specific communication boards that are developed as part of an ongoing activity. They enable us to move beyond the simple topic or choice boards which are often used in the early stages of augmentative communication development to enable the user to select a particular item or activity or topic. For example, topic boards might include choices of different foods for a meal or different toys for an activity or different activities. Topic boards do not enable the user or the message receiver, however, to make any comments about the activity or topic. Miniboards, however, provide a small evolving vocabulary centered around a particular topic or activity, designed to encourage conversation as an activity is underway. in some ways, miniboards are like the vocabulary selected for a language activity. Key vocabulary or language concepts are selected and introduced in the context of an ongoing activity.

2. miniboards can sometimes be made up ahead of time but generally are most effective if they are constructed (on the go) as part of the activity.

3. In the initial stages, certain activities or favorite games might be selected as opportunities to use miniboards. Miniboards are not intended for use throughout the day. In a school setting, two or three of the daily activities might be selected for use with miniboards. (For example, opening discussion, snack time, a particular therapy activity). At home, meal time may be a good opportunity for miniboards. With some families, meals are too hectic to use miniboards and a snack time might be an alternative. Certain play activities or times during the day when family members can select a special activity might be prime times for miniboard usage.

4. Miniboards may have only 4-6 symbols on them or they can become quite elaborate and include an extended vocabulary of 20 to 30 items. Here are some selected activities that might lend themselves to incorporation of miniboards in the activity. Examples of some of the vocabulary that might be appropriate for a given child or adult are included:

ACTIVITY	VOCABULARY
Playing with a ball and a box	ball, box, top or lid, in, out, all gone, where people's names, find, get, drop.
Building and knocking a tower over	blocks, tower, numbers, sizes, shapes, up, hit, fall down

Wind -Up figures	names of people, figures names, stop, go, fall help, stand, wind, on, off
Bowling game	pins, ball, turn. stop, fall down, yeah!, stand up too bad!
Trucks, blocks	truck, car, block, more, go, stop, CRASH, hit, names of places to drive truck or car to (e.g. Mom, store, McDonald's, grandma's).

Miniboards could also be developed for washing a doll, buying things at the store, having a tea party., playing on the farm, speace ship, going up in a balloon. In fact, any activities that the child might enjoy could incorporate the use of a miniboard. The to its successful use is selection of vocabulary that pertains to the situation and represents the child's interests.

5. miniboards enable us to talk about a specific activity, to expand vocabulary available to a board user and to talk about specific aspects of the activity rather than general comments. It is used in a turn taking mode both in play itself and in conversation. Both people are users of the communication board and active participants in the activity.

6. It usually helps to have a matrix with blank spaces so that new items can be added on the spot. If the specific vocabulary items are needed regularly, however, it may be helpful to develop a specific drawing and select a standard arrangement for repeated usage, after a few trial constructions.

7. If a set of blank matrixes are made up ahead of time, it is easier to create new miniboards. The charts should be organized in similar patterns on each chart so that the user learns that people's names will be in the same area of the chart, action words in the same area, adjectives and emotional comments in their respective areas.

8. Color coding, perhaps with a different ink for each category, will visually break up the display and perhaps speed up locating a certain item.

9. Drawings used on miniboards should be bold and simple. Resource manuals for simple line drawings include:

Mayer-Johnson
P.O. Box 1579
Solana Beach, CA. 92075-1579
• Boardmaker - Macintosh software program used to create communication displays

Picture Talk· Bruce Bollerud
3113 Atwood Avenue
Madison, Wi 53704

Don Johnston Developmental Equipment, Inc.

P.O. Box 639, 1000 N. Rand Rd., Bldg. 115, Wauconda, Illinois 60084
312-526-2682

Standard Rebus Glossary
American Guidance Service, Inc.
Circle Pines, Minnesota 55014

10. If you have vocabulary items that are used repeatedly on different miniboards, those pictures could be xeroxed and the same drawing used. In addition, it may be useful to cover these displays with contact paper. Self-sealing openings, or clear vinyl envelopes, or photo album pages are options as well.

11. The primary use of miniboards in these play activities is for conversation. Much of the learning of the symbols is carried on in the context of the activity through labelling and modeling by the communication partner for the user. This techniques is similar to the commenting done in standard oral language therapy. Short single word, two and perhaps three word or picture combinations would be used at first to comment on things as they happen. Occasional questions might be addressed to the communication board user and opportunities to make choices about who will do something next, what will be done or which item to select should be a part of the activity.

12. It is important to allow for silences within the conversation; opportunities for the child to initiate. The partners should wait and allow this to happen. If the child or adult makes a comment verbally, through bodily gestures or reaction or by pointing to something on the communication board, the partner should repeat that comment or their best interpretation of it, simultaneously pointing to the key words on the communication miniboard. If the appropriate words are not there they should be drawn in quickly at that time.

13. Rather than requesting that a chid say phrase more completely, add additional symbols to their comment, by modelling. In addition, an alternate strategy for encouraging expansion is by the user, is to point out a message which is not completely accurate as your interpretation and look for clarification from the user.

14. As the communication board user becomes more adapt at using the miniboards, it will be possible to select vocabulary items used most frequently from all miniboards and consolidate them on a primary board.

Donna DePape
Communication Aids and Systems Clinic

Recommended readings:

Bottorf, L. and DePape, D. Initiating communication systems for severely speech impaired persons. In D.E. Yoder (Ed.), Topics in Language Disorders, 2(2), 55-71. Gaithersburg: Aspen Systems Corporation, 1982.

Brandenburg, S., and Vanderheiden, G.C. (1987). Communication Board Design and Vocabulary Selection. In L. Bernstein (Ed.), Vocally Impaired- Clinical Research and Practice. Academic Press.

Calculator, S.N. (1977, October). Design and revision of nonoral systems of communication for the mentally retarded and handicapped: A discussion of the binary visually encoding board with general implications for communication. Unpublished paper, University of Wisconsin-Madison.

DePape, D. (1985). General suggestions for development of miniboards. A handout for the Communication Aids and Systems Clinic, University of Wisconsin, Madison, Wisconsin 53705.

Frumkin, J. (1987). Enhancing interaction through role playing. In S. Blackstone (Ed.), Augmentative communication: an introduction.. Rockville, MD.: ASHA.

Frumkin, J. (in press). Introduction to Augmentative Communication. Minnesota Governor's Council on Developmental Disabilities. Minneapolis, Minn.

Harris-Vanderheiden, D. and Vanderheiden, G.C. (1977). Basic considerations in the development of communicative and interactive skills for non-vocal severely physically handicapped children. In E. Sontag (Ed.), Educational Programming for the Severely and Profoundly Handicapped. Reston, VA: Council for Exceptional Children, 1977.

Harris, D. and Vanderheiden, G. (1980a). Enhancing development of communicative interaction in non-vocal severely physically handicapped children. In R.L. Schiefelbusch (Ed.), Non-speech Language and Communication Intervention. Baltimore: University Park Press.

Musselwhite, C.R. and St. Louis, K.W. (1988). Communication programming for the severely handicapped: Vocal and nonvocal strategies. Houston, TX: College-Hill Press.

Non-oral Communication: A Training Guide for the Child Without Speech. (1980). Fountain Valley School District, West Orange County Consortium for Special Education, Plavan School 9675 Warner Avenue, Fountain Valley, CA 92708.

ESTABLISHING REPRESENTATION THROUGH OBJECTS

Initially, NAME should receive instruction focusing on developing an object based communication program. Specifically, he should learn that he can use an object as a representation in order to communicate a preferance. Instruction with such a system will focus on the competencies necessary for a successful use of a more symbolic based communication system. The following is suggested:

1. Identify highly reinforcing items and obtain duplicates of the objects.
2. Identify mildly aversive or low interest items and obtain duplicates of the objects.
3. Place one reinforcing item and one low interest item in large cubes, allowing NAME to see the items but not touch them.
   Suggestions for cubes include:
   a. Clear photo cubes (5 X 8 size)
   b. Compact disk storage cases as demonstrated during today's meeting (found at Crate and Barrel).
   c. Custom made display cases
   d. Clear plastic storage boxes available through many discount stores.
4. Ask NAME "What do you want?".
5. Each time he reaches for/touches the box containing the desired object, he should immediately be provided with the duplicate of the object. thus demonstrating the representational value of the object in the box.
6. Be sure to vary the side the desired item is placed on in order to avoid perseveration to one side.

Costello, J.M., 1988

BEST COPY AVAILABLE

FACILITATING OBJECT TO PICTURE ASSOCIATION

Once NAME is clearly demonstrating representational skills and is consistent with making selections, he should be provided with instruction focusing on a more abstract level or representation. This will allow for an easier presentation of choices as well as a more functional means for a child who is independently ambulating to communicate in a variety of settings. Three potential programs follow which have been successful with a variety of people focusing on the goals of object to picture association. NAME's rate of success with previous programs may dictate which of these three is tried first.

1. As outlined above, highly reinforcing objects as well as a mildly aversive object should be placed within a clear plastic cube (available in photo supply stores) in order to emphasize the representational value and three dimensional aspects of these items. NAME should be required to point to the cube containing the desired object. The location of the items should be varied in order to avoid perseveration to one side and to ensure that NAME is attending to the item and not the location. Each time NAME reaches for the cube, he should immediately be provided with the opportunity to interact with the requested item. The item provided, however, should be a duplicate of the item in the cube, thus preserving the representational nature of the cubes. Once NAME is consistently selecting the desired items, a photograph of the items should be suspended within the cubes, taking the place of the object. When NAME is consistently selecting the desired item, the photograph should be cut so that only the item is visible and the background is removed. This preserves the three dimensional representation although the stimulus is now two dimensional. The identical program described above should be carried out at this level. Finally, the photo should be removed from the cube and placed on a flat board.
2. Two photographs should be placed on the examiner's side of a large sheet of plexiglass. NAME should be encouraged to look at these photographs while being asked "What do you want?". The actual object should be placed behind the photograph, therefore, when NAME reaches for the object, he will touch the photo. He should then interact with a duplicate of the object behind the plexiglass. In some cases, hand over hand modeling may be needed to teach NAME to point to the photograph. Following trials at this level, the object should then be placed further out of sight (i.e., several inches back from photograph). When NAME reaches for the object, hand over hand modeling should be used to re-direct him to the photograph. The object should then be immediately presented. Through this association, NAME may learn to point to the photo to receive a desired item. The location of the photos on display should be varied in order to avoid perseveration to one side. Again, as NAME demonstrates success, the object should be moved further out of sight until it has finally been removed from behind the plexiglass. If NAME begins to consistently select the photo of the desired item while avoiding the mildly aversive time, it will be clear that he is attending to the photograph and recognizes its representational value.
3. Place the container of a favored food (e.g., fruited yogurt) which NAME recognizes on the examiner's side of a sheet of plexiglass. Each time NAME reaches for this container, he should be provided with the food item (from a different container) immediately. The container should now be cut so that only the front half remains. This begins to reduce the three-dimensionality of the container. As NAME has success with this, the container should be cut down further until only the front panel remains. When NAME is able to consistently point to this front cover and avoid a less desirable choice, a photograph of the front cover should be presented.

Costello, J.M. 1988

FACILITATING INTENTIONAL COMMUNICATION BEHAVIOR

In order to become a more effective communicator, NAME should be required to actively communicate his preferances. Thus, NAME should be provided with a choice (*A*ntecedent), be expected to select a preferance (*B*ehavior) and then be immediately provided with that choice(*C*onsequence). This *Antecendent -Behavior -Consequence* or *A-B-C* paradigm may be used to teach

1. Highly reinforcing activities/objects which may be readily provided should be identified.
2. Mildly aversive activities/objects which may be readily provided should be identified.
3. NAME should be presented with two choices, one highly reinforcing choice and one mildly aversive choice.
4. The symbol set should be one with which NAME has demonstrated proficiency.
5. Choices should be clearly presented and spaced so as to facilitate clear and easily interpreted responses.
6. Choices should be clearly labeled by the communication partner (i.e., verbally state "NAME do you want _____ or do you want _____?" while pointing to the appropriate choice in order to provide visual cues).
7. NAME should then be expected to point to a choice in response to the prompt "What do you want?".
8. When appropriate, NAME should be prompted to 'look where he is pointing' in order to effectively pair the behavior with the consequence.
9. The choice object/activity should be immediately provided along with the verbal cue "You chose the _____. Here's the _____".
10. NAME should be required to interact with the choice for at least five seconds.
11. The location of choices on the display should be varied in order to be sure that NAME is attending to the task and not the location.
12. Following repeated trials over several days, if NAME clearly avoids the mildly aversive choice, it will be evident that he understands the task.

It is important to differentiate this communication program, requiring the participant to actively make choices in order to encourage functional communication skills from a language program which requires the participant to passively identify labeled objects and pictures.

Costello, J.M. 1988

Toys for Youngsters with Handicaps
• Resources •

Below is a listing of some excellent resources for toys and developmental tools for use by youngsters with handicapping conditions. It is highly recommended that selection and application be guided by a professional familiar with the child.

• Ablenet
Cerebral Palsy Center, Inc.
360 Hoover Street N.E.
Minneapolis, MN 55413
(612) 331-5958
Contact: Cheryl Waite

• Adaptive Communication Systems, Inc.
Box 12440
Pittsburgh, PA 15231
(412) 264-2288

• Don Johnston Developmental Equipment
P.O. Box 639
1000 N. Rand Road- Bldg 115
Wauconda, IL 60084
(312) 526-2682

• Prentke Romich Company
1022 Heyl Road
Wooster, OH 44691
(800) 642-8255

• Simplified Technology for the Severely Handicapped
Controls for Battery operated toys and computers.
Linda J. Burkhart
8503 Rhode Island Avenue
College Park, MD 20740
(301) 345-9152

• Steven Kanor, PhD., Inc.
8 Herkimer Avenue
Hewlett, NY 11557
(516) 783-6483

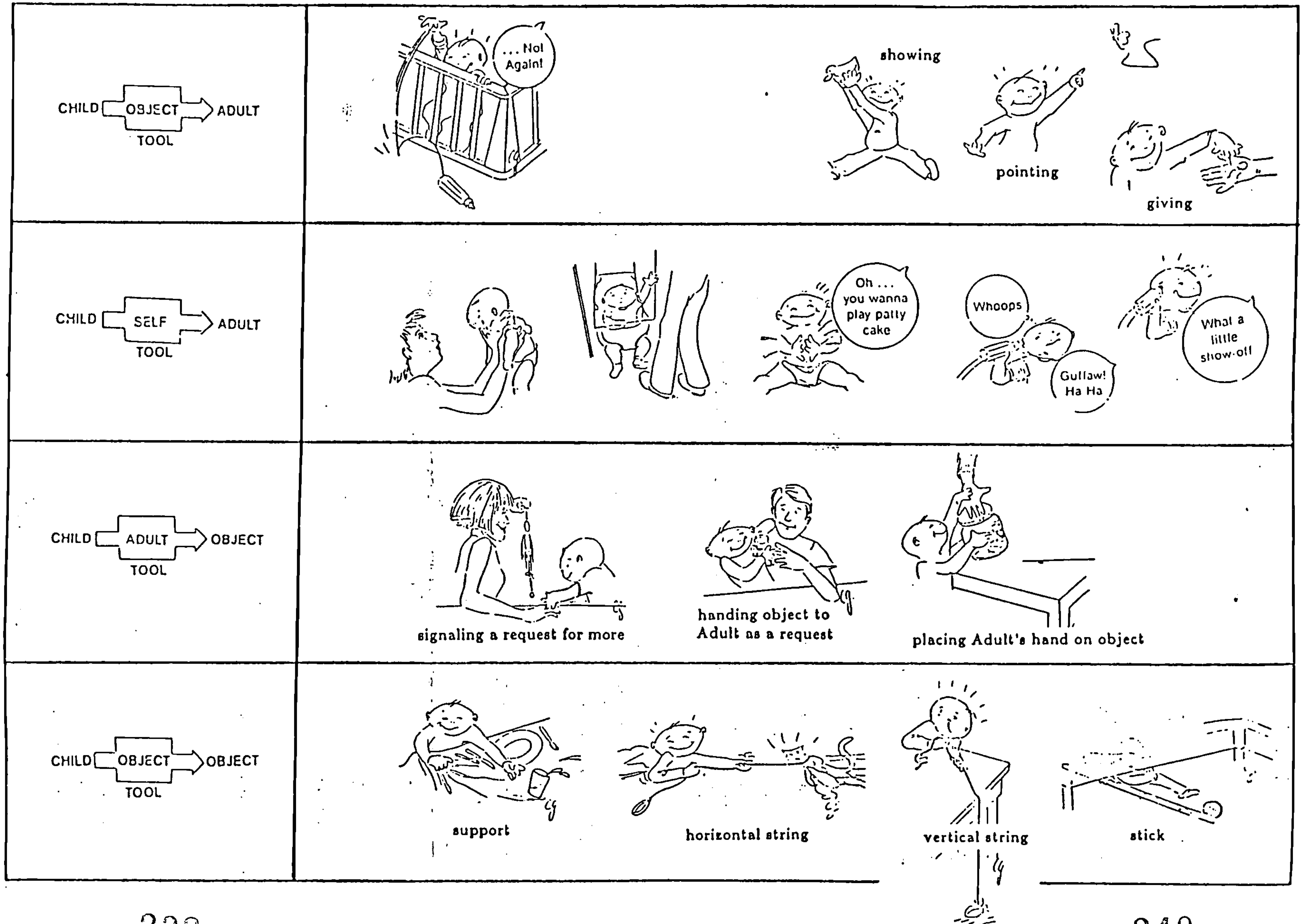
• TASH, Inc.
70 Gibson Drive, Unit 12
Markham, Ontario, Canada L3R 4C2
(416) 475-2212

• Therapeutic Toys, Inc.
91 Newberry Road
East Haddam, CT 06423
(800) 638-0676
Michael Skubel

2/89

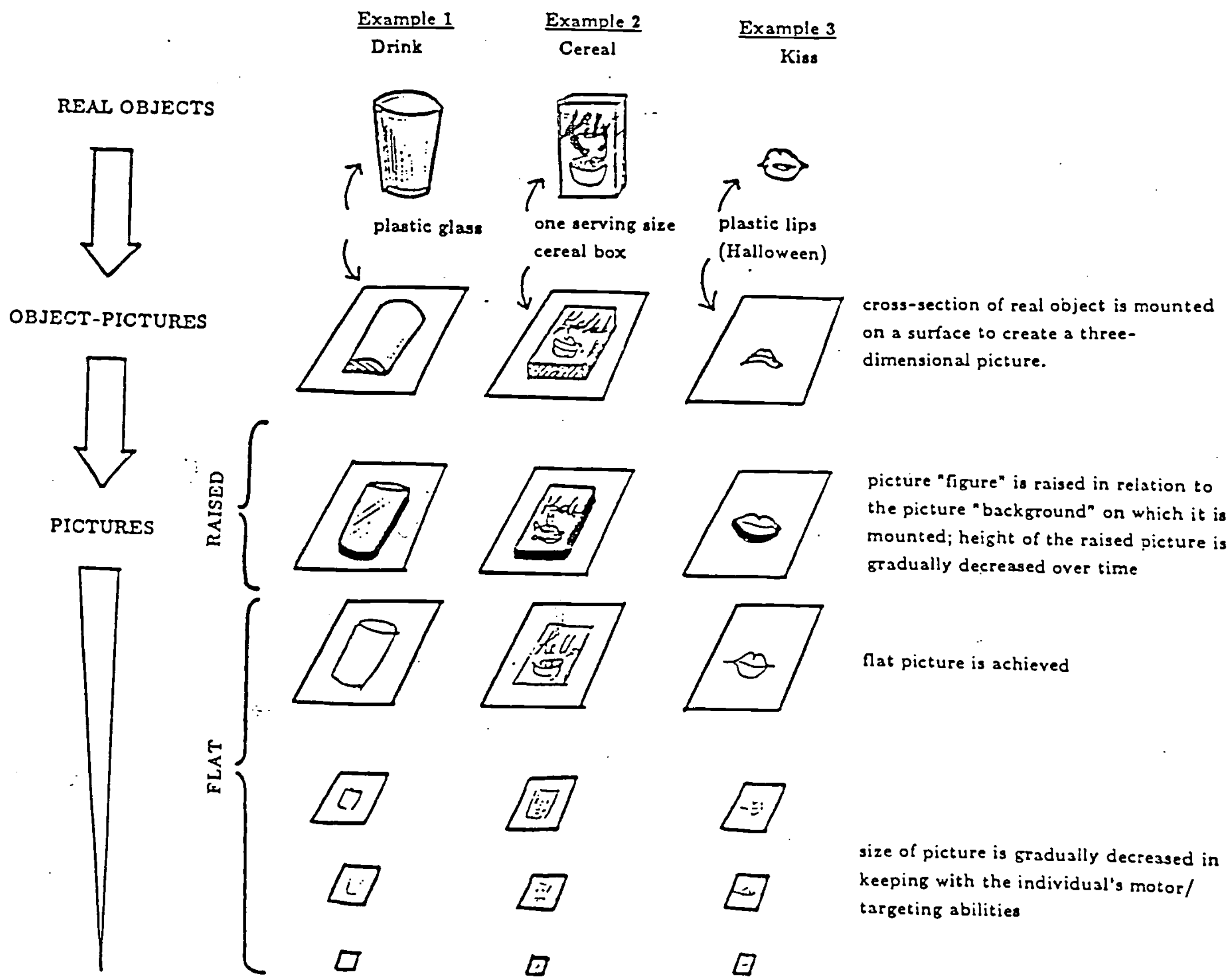
THE FOUR TOOLS: A READINESS FOR SYMBOLIC COMMUNICATION

Early Later



Envirotech and Ultra-Glo (Chemco) (available at craft supply stores) can be used to emboss miniature objects or small real objects such as Cherrios (Musselwhite, personal communication). As the plastic coating process retains the contour of the mounted miniature, this display format is well-suited for use with visually-impaired Clients.

3. Consider using "object-pictures" to effect a smooth transition from real objects to graphic representations of real objects.



ASSESSING THE ABILITY OF VERY YOUNG AND LOW-FUNCTIONING STUDENTS TO USE SWITCHES AS A TOOL FOR CONTROLLING COMMUNICATION DEVICES

Gail M. Van Tatenhove

Purpose

To assess the student's development of means-end and causality concepts relative to the use of a switch-controlled communication device.

Underlying Assumptions

Theoretical Premises

1. Functional use of an electronic communication aid involves more than the mechanical operation of the device.

2. Many teachers and clinicians expose students to the concept that switches can be used as tools by connecting the switch to a battery-operated toy. While this procedure may be a valid means to teach switch-end device association, it does not promote the communicative use of a switch-controlled augmentative communication aid. Students who recognize the switch as a means to control a toy may or may not recognize that a switch can control a communication device which, in turn, can express a message that controls the behavior of a communication partner (see Appendix A for a hierarchy of tool use relative to communication). When assessing a student who may require an electronic communication device in the future, clinicians should consider what tool-use skills need to be developed. For example, if the student needs to use an interface (e.g., headpointer), does he or she understand its use as a tool? Does the student recognize the recommended augmentative communication aid as a tool to communicate ideas, express feelings, etc.?

Client Characteristics

1. Age Range: 12 months to 2 years (cognitively).

2. Developmental Level: Infant, preschool.

3. Disability Type: Orthopedically impaired, with or without learning impairment (MR/LD); mildly/moderately/severely intellectually impaired; autistic.

Setting: Evaluation center.

Augmentative Aids, Symbols, and Techniques

Low-tech aids: Battery operated toys with a variety of single-action switches.

High-tech aids: Simple scanning device (e.g., Zygo 16, Versascan,[1] Light Talker with single-icon messages[2]).

Techniques: On/off; single row/column scanning.

Description of Strategy

A. Selecting an Appropriate Switch

1. Interview the student's occupational therapist, parent, and teacher to determine previous experience using a switch to control equipment or appliances, such as a remote control for the television. Observe, if possible, the student's use of a switch.

2. If the student does not have previous switch experience, consult with the occupational or physical therapist to determine the student's best access mode (i.e., reliable body part and kind of movement to use).

3. Select a switch that has a low cognitive demand using the following hierarchy as a guide:

Easiest	to	Hardest
Student must see the connection between the switch and the result (e.g., tread switch activated by hand with the cord visible).	Student requires some feedback from the switch but does not need to "see the connection" (e.g., tread switch with auditory feedback activated by body part he or she cannot see).	Student can use a switch with limited feedback, that has no obvious connection to a device, and is activated by a body part not seen (e.g., light pointer).

B. Setting up an Appropriate Switch and End Device

1. Select an on/off device (preferably a simple toy) that will not be overly stimulating to the student (e.g., a tape player that is on when the student activates the switch and is off when the student releases the switch).

---

[1]Available from Zygo Industries, P.O. Box 1008, Portland, OR 97207.
[2]Available from Prentke Romich Co., 1022 Heyl Road, Wooster, OH 44691.

2. Make whatever compromise is necessary between the access mode and switch/device selected to provide the best set-up with the least cognitive demand. For example, a student with severe cognitive as well as physical limitations may need a switch by the head. However, if the student cannot see when the switch is activated, the cognitive demand of the task may be too high. Therefore, the OT or PT may need to assist the student in activating the switch to turn on or off the device. Caution: The OT/PT must wait and feel the student make a volitional movement before helping the student to activate the switch.

3. Once the switch and end device is set up, allow the student to play and experiment with it. Usually two to five minutes are adequate.

C. Assessing Potential Switch Use for a Communication Aid

After students have experience with the switch and end device, analyze responses according to the following scale:

Response #1 - student focuses attention (measured through eye gaze) on his or her action on the switch, appearing continually surprised at the result on the end device.

Response #2 - student alternates gaze between the action on the switch and the resulting effect on the toy, appearing to develop a beginning association between the two.

Response #3 - student maintains attention on the end device, disregarding his or her action on the switch.

If the student consistently repeats response #1 (maintains gaze on switch on 75% of trials), continue switch-to-toy and switch-to-recreational computer experiences. Reassess after continued experiences with a switch for recreational purposes to determine when the student can use a switch to control a communication device.

If the student demonstrates response #2 or #3 on 75% of the trials, evaluate the student's ability to use a communication device.

D. Facilitating Switch Use Understanding

Switch-end device use can be facilitated, if necessary. For example, if a student simply activates a switch to control an on/off toy, significant goal-directed behaviors are not involved (i.e., action/no action is all the toy can do). A different toy may permit the child to understand a more advanced concept. For example, a toy bear that climbs can be made to climb to a designated location. In this case, the student must understand the goal and activate the switch until the goal is achieved.

By changing variables involved in switch-end device use, switch use can be enhanced. For example, change the

1. end device; maintain same mode of operation (e.g., simple on/off).

2. type of switch; maintain the same mode of operation (e.g., momentary single action[3]).

3. access site of the switch (i.e., move the switch to a second, approprate body part) for a more timely, reliable access, if necessary.

4. type of switch and the mode of operation (e.g., dual action or latching switch[4]).

5. end device and the mode of operation (e.g., a goal-directed toy that requires repeated activation of the switch).

The student's responses to changes in these variables permit the facilitator to determine at what level to begin switch activation training. For example, if the student continues to exhibit response #1 after all variables are changed systematically, the facilitator will provide additional experience at that level. The student would not be ready to use a switch-controlled communication system. Continued recreational use of a switch with a variety of end devices would be required.

When the student demonstrates responses #2 and #3 when presented with variables described, the student's ability to use a specific communication device in a functional activity may be evaluated.

E. Evaluating Functional Use of a Communication Device

1. Use a simple step-scanning device with a momentary action switch. Use the switch access mode that has proven best for the student. Place a stimulus item, such as a small object or photograph, in a target area. Functional and fun objects, such as wind-up toys, balloons, and bubbles, work well.

2. Ask the student to advance the scanning light to the designated target area. When the light reaches the target, give the student the object as a meaningful reward.

3. Place a second, less motivating stimulus item in a second target area (keeping the first item in the original target area). Ask the student

---

[3]A momentary single action switch is similar to a doorbell. It has an on-off action.

[4]A dual action or latching switch turns on when activated and remains on when released.

to choose between the two objects by moving the light to the desired object. Give the student the object selected.

4. Record the amount of prompting and guidance necessary for the student to use the device in a functional, communicative manner, including: (a) the cues needed for the student to understand that he or she needs to advance the light to the designated target; (b) whether the student directs eye gaze to the examiner or vocalizes upon reaching the desired target area; or (c) whether the student reverts and no longer understands the relationship between the switch and the end device.

5. Using clinical judgment, provide additional trials, add additional stimulus items, vary the location of stimulus items, provide additional practice time, etc.

6. If the student does not demonstrate functional, interactive use of the communication device, continue to use switch-controlled, recreational devices. Consider use of a communication device only in a controlled treatment setting using activities that will promote communicative use and understanding of the device.

7. Use a switch-controlled communication device in a variety of settings if the student demonstrates functional interactive use of the device. Provide interaction guidelines to professionals, parents, siblings, and peers working with the student.

Evaluation of Effectiveness

The procedure was validated clinically. Approximately 50 students were screened over a six-month period. An analysis of the responses of five students at each of the three response levels was made to compare the type of responses observed and each student's functional use of a communication device. Then, one student at each response level was selected and followed for 14 months. Results were consistent in that each student progressed through understanding the use of switches as outlined.

Appendix A

HIERARCHY OF TOOL USE RELATIVE TO COMMUNICATION

ACTS INDEPENDENTLY: noncommunicative

- uses body - grabs, goes, gets, looks at

EXTENDS USE OF BODY TO GET ANOTHER TO ACT: communicative

- points to object and looks at person
- looks at object, then to person, and back to object
- takes hand and shows

USES TOOL TO GET ANOTHER TO ACT: communicative

- looks or points at object/picture on device and looks at partner

USES TOOL ON TOOL TO GET ANOTHER TO ACT: communicative

- uses headpointer, T-bar, or light source on head to point to symbols on the device and, thereby, communicate ideas intentionally

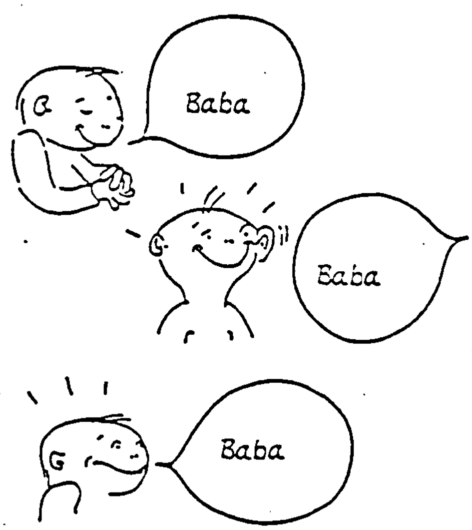
SUGGESTIONS FOR STIMULATING VOCAL IMITATION

Vocal imitation skills lay the groundwork for later conversational turn-taking. When attempting to establish vocal imitation skills, care should be taken to follow the normal developmental sequence. The following are suggestions for encourgaging the development of vocal imitation skills.

1. Make a conscious effort to playfully imitate sound(s) that your Child has just produced. Be sure to make it a game. At this time, your Child may have several sounds/sound sequences within his/her repertoire that you can imitate. Any sound that your Child produces can become the focus for vocal turn-taking.

   Examples:

   Child: Baba*
   Caregiver: Baba
   Child: Baba
   Caregiver: Baba, etc.

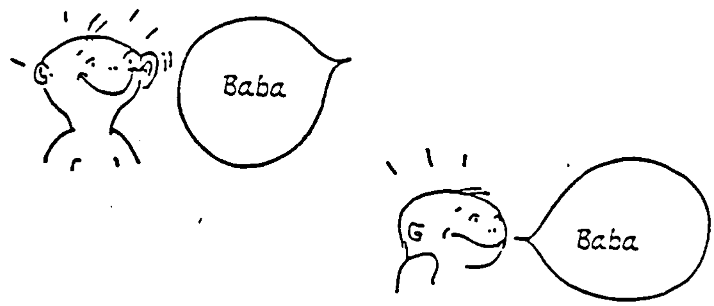


2. With time, you should be able to start the turn-taking sequence using sounds/sound sequences that your child frequently produces (i.e., familiar sounds).

*The examples used in this program are only examples and may not be sounds within your Child's repertoire.

Examples:

Caregiver:	Baba (knowing this is a sound sequence that your Child frequently produces)
Child:	Baba
Caregiver:	Baba, etc.



3. Later, when the Child is able to reliably perform suggestion 2 for several sounds/sound sequences in his/her repertoire, try switching off between different sounds in his/her repertoire.

Examples:

Caregiver:	Eeee (knowing this is a sound sequence that your Child frequently produces)
Child:	Eeee
Caregiver:	Eeee
Child:	Eeee
Caregiver:	Baba (knowing this is a sound sequence he/she frequently produces)
Child:	Baba
Caregiver:	Baba, etc.

4. Much later in intervention when the Child is able to reliably perform suggestion 3, try switching off to new sound sequences (i.e., sounds that are not currently in his/her repertoire, sounds you've never heard him/her produce before).

Examples:

Caregiver:	Eeee (a sound sequence that your Child typically produces)
Child:	Eeee
Caregiver:	Eeee
Child:	Eeee
Caregiver:	Boo-boo-boo (a sound sequence that your Child does not typically produce)
Child:	Ba-ba-boo

NOTE: Initially the Child's imitations may not be entirely accurate. With time, however, he/she will acquire greater skill in matching his/her vocalization with yours.

5. At this point, the Child should already be attempting to imitate some of your words.

Handout developed by:

© Carol Goossens', Ph.D., CCC-SP
Sharon Crain, M.S., CCC-SP
Sparks Center for Developmental and Learning Disorders
University of Alabama at Birmingham
P. O. Box 313, University Station
Birmingham, Alabama 35294

Augmentative Communication News

July, 1989 Vol. 2, No.4

INSIDE THIS ISSUE . . .

UPFRONT

This issue focuses on Visual Scanning. I was fortunate to speak with many individuals with expertise, information and exciting ideas about the topic (see Resources). Some interviews were conducted in New Orleans at the end of June while attending the first meeting of the U.S. Society of Augmentative and Alternative Communication (USSAAC) held in conjunction with RESNA. Very stimulating, fun, and *HOT* in New Orleans!

For Consumers attempts to relate what is known about how able-bodied humans "learn complex motor skills" to the experiences of those attempting to develop visual scanning techniques.

Clinical News focuses on issues and practices relevant to teaching visual scanning techniques. In **Equipment**, you'll find a brief description of available scanning communication aids and computer software that can facilitate the development of various types of scanning skills.

In **University/Research** ongoing research activities in Sweden are highlighted. For those planning to attend the 4th International Augmentative Alternative Communication Conference in Stockholm next August, the article may help you plan your August, 1990 "working" holiday in Sweden.

The **Governmental section** provides a brief update on the status of the Tech Bill in the U.S. *(cont. pg. 2)*

For Consumers

Visual Scanning: What's it all about?

Individuals who use visual scanning techniques typically have severe motor impairments that preclude and greatly restrict their direct access to communication devices and computers. Included are those with cerebral palsy, degenerating neurologic conditions, traumatic brain injury, and others.

Although visual scanning does indeed provide access, it is notoriously slow. Therefore, some consider it a choice of last resort. However, visual scanning can be and is a feasible option - worthy of consideration.

- Visual scanning may be less fatiguing motorically than direct selection techniques. For some, it is faster.[1]
- Visual scanning may be accomplished in positions where direct selection access to devices is impossible, e.g., lying down.
- Many software programs and communication aids have visual scanning options.
- It's not an "either/or" proposition. All individuals should use multiple techniques to participate in communication, education, leisure, and vocational activities and to control their environment.

Visual scanning is a skill, and skills are learned.

Skill Development

The development of motor skills is a complex process. Most of us haven't given much thought to what's entailed. *Try to recall learning (or trying to learn) to ski, drive a car or wheelchair, type, use a light pointer, play racquetball, dance, or operate a communication aid to converse with a friend!* Because adults avoid learning new motor tasks, we tend to forget how complicated and difficult it can be.[2]

Models of complex motor learning can help professionals and consumers think about tasks like visual scanning. Robb[3] *(cont.pg.2)*

Upfront (from page 1)

Lots of Hotline calls this month: service delivery & public policy concerns, questions about aphasia software, requests for ideas re: clients. Whatever questions you may have, just call (408) 649-3050. I'll try to help. *Please keep in mind, we are on Pacific Standard Time.* Thanks!
We hope you are enjoying the summer (or winter for subscribers down under.) *We're planning a trip to New Zealand & Australia in September.* Remember, don't take those electronic communication aids in the water. . . swimming!

Visual Scanning (from page 1)

characterizes skill development as a process, which occurs in 3 phases:

Phase I - Plan formation: Initially, individuals must learn *what is to be done and why*. Phase I can take a few minutes, hours, days, or even months. This varies with the complexity of the task (e.g., type of scanning) and with the learner's capacities and limitations.

Phase II - Practice: The key in Phase II is for learners to engage in *meaningful, appropriate practice with feedback*. The amount of practice needed varies with the complexity of the task and the past experiences and capabilities of the individual. The literature suggests able-bodied persons of different ages approach practice differently. Children are curious and easily challenged; they *practice* to "see what they can do." Adults want results quickly and hope to achieve without putting forth much effort.

Feedback is the single most important factor. Simply doing something over and over again does not improve performance. *We sign our names again and again, but I'll bet your signature hasn't gotten any more legible than mine over time!*

Feedback can play 3 roles: *motivate*, *reinforce*, and/or *regulate* behavior. Errors occurring during practice are necessary for learning. However, without meaningful feedback (i.e., what caused the error and how to correct it), the learner is probably not going to progress. Simply giving positive reinforcement (e.g., saying "good job," or displaying "blinking blobs") may keep learners motivated and working at a task, but will not help them develop and refine skills! *"Teachers who develop a discriminatory eye for detecting errors in sequential and temporal patterning are the best source of feedback."*[4]

Phase III - Automatic execution: The last phase begins when the learner performs the total movement pattern with fairly consistent results. Stress and anxiety are reduced because tasks have become easier. Constant monitoring continues, but is relegated to subliminal levels. During this phase *complex skills become automatic, and individuals can concentrate on other factors* (e.g., communication, the content of a term paper, implementing rate enhancement strategies, etc.)

Research suggests that making major changes during Phase III, (e.g., changing the order of subroutines or introducing a different subroutine) is very difficult for able-bodied individuals to accommodate. This should make us cautious about presuming some sort of hierarchy to teaching visual scanning skills, (i.e., first introduce step scanning, then linear, then row/column, group/item, directed scanning.) To date, however, no evidence exists that training one type of scanning facilitates or interferes with learning others.

Capacities/ Limitations of Man

Information about human processing capabilities and limitations that may affect the acquisition of visual scanning skills is considered below:

1. **The sensory capacities of learners** (i.e., depth perception, peripheral vision, acuity) are generally not improved through practice. Comment: *Many individuals with severe motor problems have visual acuity, perception, field cuts, and/or fixation difficulties. This will profoundly affect their ability to use visual scanning. We often must look for ways to compensate for, rather than remediate these problems.*

2. **To detect signals,** humans must distinguish the figure from the background. Loud noises, novel stimuli, and verbal praise can heighten the ability of humans to detect signals. Comment: *The stimulus (e.g., cursor, LED, pointer, target) should be the most obvious thing around. Other aspects should be minimized during training. Learners must be able to maintain their gaze on targets while they activate switches.*

3. **Selective attention** is important to the development of complex skills. Humans can (and do) filter out irrelevant information. Comment: *Persons with central nervous system problems and young children often have difficulty filtering out "irrelevant" stimuli (e.g., "attention problems," "distractibility," "hyperactivity.") Setting up contexts that allow learning to occur is critical.*

4. **Perception,** i.e., interpretation of sensory information, is influenced by instructions and an individual's experience with a task. Also, feedback affects perception (see Phase II). Comment: *Individuals who are learning new skills need all 3 types of feedback.*

5. **Vigilance tasks** depend on a person's monitoring capacities. Monitoring capacities are dependent on how much, how fast, how predictable stimuli sources are, as well as on the memory capacity of individuals. If events occur slowly, learners can more easily recognize and process sensory input. Tasks requiring vigilance are acquired more easily with frequent rest periods.

AUGMENTATIVE COMMUNICATION NEWS is an independent, non-membership publication. Copyright 1989 by Sunset Enterprises. Reproduce only with written consent.
Author, Sarah W. Blackstone, Ph.D.; Publisher, Gary K. Poock, Ph.D.
Published by Sunset Enterprises, One Surf Way, Suite #215, Monterey, CA 93940
Subscriptions: By personal check U.S.$37 (1 yr), U.S.$67 (2 yrs), U.S.$97 (3 yrs). Single issues $8.
Overseas - U.S. $47 (1 yr.), U.S. $85 (2 yrs.), or U.S. $123 (3 yrs.).
Institutions, libraries, schools, hospitals, all others: Add $20 per year. Single issues $10.
Special rates for consumers and full-time students now available.
0897-9278 Telephone (408) 649-3050

Comment: Instructions, demonstrations, and practice sessions must be controlled. Long practice sessions may not be productive. Rate enhancement techniques require vigilance.

6. Reaction time (i.e., time taken to initiate or begin a movement) is affected by learning, anticipation, and the amount of information people must process, as well as their motor capabilities. The more choices someone has, the longer the reaction time. *Comment: Visual scanning arrays typically contain an enormous amount of information. If someone must search for targets, reaction time will increase. Dynamic arrays and encoding techniques also increase the amount of information processing required and will affect reaction times.*

7. Movement time (time between initiation and completion of movement) is affected by the need for accuracy. Able-bodied humans can and do trade off speed and accuracy. *Comment: Activating a switch involves reaction and movement time. Thus, motor planning plays a major role in the speed and accuracy of visual scanning skill development.*

Performance of Able-bodied Persons on Scanning Tasks

Direct selection vs. scanning. Ratcliff[5] compared the accuracy of able-bodied children using direct selection and scanning techniques on the Light Talker (1st through 5th graders) on a direction-following task. Results indicated a significantly *greater number of errors for all those using scanning.*

Visual vs. Auditory Scanning. Fried-Oken[6] compared the performance of 90 adults using 3 scanning approaches on a sentence recognition task: visual only, auditory only, and visual/auditory. The task was designed to simulate typical conditions for communication device users. Results showed *response accuracy and reaction times were significantly better for the visual scanning only condition*. Auditory scanning was the poorest and combined scanning fell between the two. Makes sense!

Clinical News

Visual Scanning: Training Approaches

Visual scanning is the sequential presentation of choices (letters, words, or pictures). As options are presented by a person or machine, the user signals (i.e. looks up, vocalizes, pushes a switch) just at the "moment" the desired item (or group, which contains the item) is highlighted. Master teachers, clinicians, researchers with whom I spoke, agree visual scanning techniques involve multiple skill components: motor, motor planning, visual, perceptual, kinesthetic, cognitive, and linguistic. Prior to introducing visual scanning techniques, certain assumptions are made:

1. *Individuals are positioned in ways that allow and facilitate function.* Seating specialists often focus on adaptive seating for wheelchairs; however, communication occurs everywhere so other positions are considered. Steps toward positioning the body to augment function are summarized in Table I.

Table I. Positioning for Function

STEPS	STRATEGIES TO CONSIDER
1st Stabilize pelvis (normalize tone, & reflexes)	Seat belt; Type of cushion; Contour and angle of insert (e.g., allowing gravity to assist) Angle at hips (generally 90°),
2nd Stabilize trunk	Midline orientation: Shoulder harnessing; Strapping; Lap tray, Side supports
3rd Stabilize arms	Arm restraints; Protracter pads Lap tray adaptations, Dowels
4th Stabilize lower extremities, feet	90° at ankle and knees Orientation of insert Ankle straps; shoe cups; calf straps; toe loops; foot rests

2. *An appropriate interface is established.* Professionals with experience and expertise in assessing the human/machine interface (e.g., rehabilitation engineers, O.T.s, some educators and speech-language pathologists) make these decisions in consultation with the individual and family. Anatomic sites under the control of the individual are identified, and movement patterns described. Abnormal reflexes often are a major consideration. Finally, interface/switch options are assessed and selected. Assessment software can be helpful.[7] Table II lists questions to get answers to.

Table II. Questions to consider

Does the person activate the switch? How long does it take? Can the person hit without prehits or multiple activations. Can he activate it at a particular time? Does he stay on the switch? How long? Can he stay off the switch? Can he release the switch on verbal command? on a visual target? Can he coordinate the timing of activation and release with the visual components of task, i.e., tracking, fixation, etc.? Can he see the display and activate the switch? Does he fatigue?

3. *The position and stability of the switch and the display is optimized.* This is an often overlooked, yet critical step. Visual, perceptual and motoric factors are pertinent. Decisions are made re: how far away and at what angle the switch(es) and visual display is placed and mounted. Ways to mount and stabilize switches are available from commercial manufacturers/distributors. Get some help! Goossens' and Crain in their wonderful new book describe success using chlorinated polyvinyl chloride (CPVC) pipe mounts.[8]

4. *Major alterations in the position of individuals, switches, and the device/display are completed prior to implementing high technology.* Scanning training often begins with no tech or low tech approaches while positioning/seating and interface decisions are being made.

5. *If individuals do not understand the task of scanning, yet need a means of interacting with some independence, a conceptual bridge can be built through instruction.* To do so, increase a person's familiarity with the task demands by maximizing important features, presenting important sequences and relations in a coherent whole, & minimizing information load by stripping the task of all but essential elements.[9]

The level at which individuals can accomplish different scanning techniques is unknown. Some say 2 year olds can use switches to make simple choices. Normal 3 year olds can use the Versascan *(cont. pg. 4)*

with remote lamps and the Zygo 16.[9] Light's[10] research shows linear scanning requires two coordinated motor/cognitive operations (monitoring the relation of the cursor movement through a visual array of pictures and timing activation of the switch to make a selection) and concludes it is probably a 4 year level task that can be taught to those younger.

- She analyzed the performance of 3 children (experts) who successfully used automatic linear scanning to access a computer (2 were able-bodied, 1 was nonspeaking and severely physically disabled.) Errors were similar, i.e., anticipating the cursor's movement and activating the switch prematurely. The 2 able-bodied children visually tracked the cursor, while the physically disabled child focused on the target, a more advanced strategy.

One thing is for sure. Clinicians disagree. Some report 3 1/2 year olds learn row/column scanning. Most report success only for those above 4 years old (developmentally). Few ventured a guess for directed and block scanning...maybe 7 year olds? Researchers! We need to know when to build bridges and how.

Task Analysis

In the past, professionals *(I'm guilty)* taught visual scanning techniques without first analyzing the task or thinking beyond its motor components. Steps to a task analysis are:

- 1. Always try the task yourself.
- 2. Ask others who have mastered it, how they do it? Observe what they do.
- 3. Break the task down into the motor, visual, cognitive, linguistic, social requirements. Then, consider how components are coordinated.
- 4. Develop a training strategy based on what you know about the equipment (i.e., switch, software, hardware) and have observed about a client.
- 5. Implement your strategy, evaluate the effects, and make modifications.

Training Guidelines

The 3 Phases described in *For Consumers* provide the framework for this section. Assessment is treated as an integral and ongoing part of the process of skill development so is not considered separately.

I. Understanding Scanning

Phase I emphasizes cognitive factors, minimizing motor, linguistic, social, and visual components. Whereas young and mentally handicapped individuals may require months of training, older children and adults will catch on after a brief demonstration and description.

- Demonstrations Showing the sequential ordering of subroutines and giving information about the purpose or objective of the task.
- Descriptions Communicating about what learner is to learn (e.g., I'm pushing the switch, I'm watching the target. Here comes the cursor. There, I got it!).

Because visual scanning skills develop at a young age without technology (e.g., watching people, locating and looking at desired objects, making choices), we often begin with a "low/no tech" emphasis. For example:

- Linear scanning of objects, pictures, people. "Let me know...do you want this, this, or this?" Individual indicates "Yes" as you point to choice by smiling, vocalizing, looking up, touching a buzzer, turning toward a red pad, etc..
- An Etran or Eye-Gaze Vest which provides communication and language training also can be used to introduce visual scanning techniques (i.e., locating targets, making selections.)
- Linear, row/column or block/item scanning using symbol, word, or alphabet boards can easily be introduced as listener assisted techniques.

Few have difficulty understanding what a switch does. If there is a problem, consider the following:

- VanTatenhove[11] describes 3 levels of responses from young and severely mentally handicapped children: a) Child focuses on switch with limited, if any, awareness of its relationship to anything else; b) Child focuses both on switch and the "effect" with awareness of the relationship. c) Child sees switch as a way to make things happen and focuses on the goal (e.g., making screen change; advancing slides in projector, activating a loop tape to call friend).
- When switches are a distraction, make the consequence of switch activation (e.g.,visual display, fan) more obvious and the switch invisible. For example, use a computer and turn off the lights.

When you plan to introduce a specific device or computer program that uses scanning, adults and older children will readily grasp the task. For younger children,

- games, play activities, and familiar routines provide learning contexts. Computer software and training aids can be very helpful (see Equipment).[12,13,14]

If you can't tell what someone understands, Cook[15] describes a protocol that can help:

Observe motor behaviors in response to: a) non time-contingent switch activation (prompt free); b) hold/release activities; c) time contingent switch activation; d) hit/release activities; and e) choice making.

II. Practicing Scanning

Everyone needs lots of practice in Phase II. Adults, like children, require carefully thought out intervention approaches to skill development.

In Phase II, tasks with low cognitive/linguistic/visual loads are used to allow motor and motor planning skills to develop. Clinicians monitor fatigue and positioning of the individual, switch, and display before, during, and after practice sessions. Even small changes can affect performance.

- Teach clients to go to a "home base" between switch activations. Make it a real resting spot.[8]
- Help individuals locate and/or fixate on targets by pointing, highlighting with a light, blocking out other parts of the display, and/or providing verbal, auditory and visual cues.
- Give individuals time to experiment.
- To deal with anticipation problems, instruct person to focus on the target. Slow down the scan. Blocking out parts of display completely, at first, then gradually making parts more and more transparent may help.
- Provide meaningful feedback. This requires watching what individuals do and analyzing their behaviors, especially their errors. For example, Light[10] describes a seemingly random pattern of switch activations, which, on closer analysis, revealed that switch activations occurred each time the facilitator spoke, with little regard to the cursor position on the screen. Further analysis of the problem showed the person could indicate with a "yes" response when a target picture was pointed to and could push a switch in response to a visual cue, but had difficulty combining these operations.
- Use a simple training device with voice output, if possible. Person can select clothes for a doll, trucks for race, ingredients for a cake, or what lipstick to put on you (e.g., green, purple).
- Use peers to model switch use and play games.

How we arrange the visual display will help determine the success of training. Table III provides guidelines.

Table III. Optimizing Scanning Displays

- Place targets at individual's midline (note: may be to the right or left of "true" midline)
- Ask individual and vision specialists how far away and what size displays/symbols should be. What individual does in order to see. Is color important?[16]
- Consider placement of most frequently used targets (vocabulary) with regard to cognitive, motor and language demands. How will individual sequence language?
- Figure out which are easiest to most difficult positions to access on an array. Start by asking person to get to the easiest cell on the array. Never begin with row 1, column 1. Be very systematic about how you progress.
- Think of the future (e.g, will number of locations accessed increase or decrease?).
- Take into account the type of scanning and whether or not encoding or other rate enhancement techniques will be overlaid on the scanning task.

In the beginning, speed and accuracy are of little importance. However, once an individual can accomplish the task with (cont. pg. 5)

Table V. Communication programs

Communication Enhancement Center, Boston, MA

Target $75 Trains linear, row/column, stepped, directed scanning. 4-49 locations.

Scan & Speak $75 Adds memory capability to Target.

Rehabilitation Engineering Center, Palo Alto, CA

Step By Step $100 Choice making, 9 cells.

ENABLE Syracuse, NY

Magic Symbals, $150 Semantically organized pictorgraphic symbols. Construct phrases

full of good clinical, as well as technical ideas. The Talking scan option is very powerful.

Table V lists some communication programs available for the Apple II family. They provide individuals with exposure to visual scanning techniques as a means to accomplish communication tasks.

Communication Devices with Visual Scanning Options

Manufacturers offer an increasing number of visual scanning devices. Single dimensional devices are typically considered "training or educational aids" because of vocabulary restrictions and portability factors. Multidimensional devices are commonly chosen to serve the conversational, computer access, writing needs of individuals who are unable to use more direct methods. Tables VI and VII take a snapshot of what is currently available.

Governmental
Update from Washington

States in the U.S. wishing to be granted monies alloted through P.L. 100-407 to implement an assistive technologies program for their citizens await the results of the competition. Awards will be announced sometime this summer, early fall. Also a technical assistance contract will be made soon to an organization/agency to assist states in implementing their plan. Cohen, Rehabilitation Program Specialist, NIDRR cautions states not receiving grants to "'get ready." Year 2 competitions will be coming soon. Call Cohen at (202) 732-5066.

Table VI. Single Dimensional Visual Scanning Aids

PRODUCT	COMPANY	PRICE	TYPE	SWITCH	SPEECH	PRINT	FEATURES
Clock Communicators	Steven Kanor, Inc. 8 Main St., Hastings-On-Hudson, NY 10706	$60-76	Rotary	1	No	No	Pointer; Adjustable speeds; Several available with music and size option
Dial Scan	Don Johnston Devel. Equip. P.O. Box 639 Wauconda, IL 60084	$195	Rotary	1,2	No	No	Pointer; Transparent, allowing face-to-face/ double--sided pictures; Clockwise/couter-clockwise; Variable speeds
Versascan	Prentke Romich Co 1022 Heyl Rd, Wooster, Ohio 44691	$799	Rotary	1,2	No	No	16 removable lamps, also remote lamps Translucent overlays, colored lamp covers
Sequential Scanner	Steven Kanor, Inc.	$71-86	Linear/step	1	No	No	2-4 compartments which light
Scan Wolf	AdamLab, 33500 Van Born Rd. Wayne MI 48184	$375	Linear/step	1	Yes	No	6 x 6 (36 selection areas); 30 levels Speech programmed at factory
Zygo 16C	Zygo Industries, Inc. P.O. Box 1008, Portland OR 97207-1008	$1145	Linear	1	Yes w/Parrot	No	4 x 4 matrix
Sleeper	Zygo Industries, Inc.	$1445	Omni directional	1-2	Yes	No	Backlighted, 4 x 4 matrix, Display can be angled 30° and 60°

Table VII. Multi Dimensional Visual Scanning Aids

Zygo 100	Zygo Industries, Inc.	$1950	Row/column	1	No	No	Limited memory, 16 messages
PACA	Zygo Industries, Inc.,	$1250/ $2500	Linear Row/column	1,2	Optional	Yes 24 col.	Literacy required Conversation note taking, writing, and calculation Predicts based on frequency of use
Switchboard	Zygo Industries, Inc.	$1495	Linear Row/column Directed	1-5	No	No	4 to 49 locations; overlays for several options
Macaw (scanning)	Zygo Industries, Inc.	$1495	Linear Row column	1-4	Yes	No	60 seconds high quality digitized speech 120 seconds standard speech
scanWRITER	Zygo Industries, Inc.	$4.200	Row/column Directed	1-5	Optional	Yes 20 col.	Various message storage options; calculator; Interface to computers, Literacy required
Tetrascan (keyboard emulator only)		$2250					
Light Talker	Prentke Romich Co	$4290	Linear, Row/column., Directed	1-5	Yes	Optional	Express/Minspeak firmware w/rate enhancemen 8, 32, 128 locations
Talk-O	Tash Inc., 70 Gibson Drive, # 12 Markham,Ont.L3R4C2	$2,300 + access option	Linear Row/column Directed	1-5	Yes	No	Multiple access options purchased separately, 120 seconds of recorded speech
Scanpac (Evalpac)	ACS 354 Hookstown Gr. Rd Clinton, PA 15206	$3.695 $3995	Linear Directed Block/item	1-5	Yes	Yes	Rate enhancement techniques.; 15 memory levels
WSKE II	Words +, Inc. distributed by ACS above	$3395	Row/colum Directed	1-5	Yes	Optional	Prediction, abbrev. expansion, Literacy. keyboard emulator. Datavue Spark
EZ Com	Words +, Inc	$3395	Row/column	1	Yes	Optional	Designed for visual problems. Datavue Spark
Equalizer	Words +	$3395	Row/column Directed	1-5	Yes	Optional	Datavue Spark. Literacy. Prediction learns user's patterns. Calculator, drawing, games, music
Special Friend (scanning)	Shea Products, Inc. 1721 W. Hamilton Rd. Rochester Hills, MI 48309	$3250	Row/column	1	Yes	No	26 programmable pages

some degree of accuracy, begin increasing speed, or the number of items in the array, or decrease the size of items. Change one thing at a time so you can determine its effect!

III. Using Scanning to Accomplish Tasks and Achieve Goals

Relying on scanning to communicate, complete homework assignments, requires being able to scan fast, without thinking about it. Automaticity (seen in touch typists, drivers, musicians, athletes) is needed. Speed and accuracy are major factors during Phase III, which never ends.

- Accuracy: Some feel an accuracy level of at least 80% is necessary for operational competency in scanning. How accurate should someone be before you increase speed? We don't know! Keep in mind, however. that even able-bodied people are not 100% accurate using row/column scanning techniques.
- Speed: Cook suggests thinking about thresholds rather than absolute values (i.e.. range between the fastest and slowest rate at which client can select messages. Kraat reminds us when we increase rates, we should provide clients time to accommodate to the increase.
- Fatigue factors change during a day. Give options to alter speed of scan.

Scanning is a means to an end....not the END. Ultimately, scanning techniques provide independent access to tools and enable individuals to participate in activities, interact, and accomplish a range of tasks more independently. Be patient... it takes a long time to develop complex skills.

News

Hyper ABLE DATA for Macintosh computers with 20 megabytes on a hard disk will be available soon. It is very, very nice. Assistive device products and companies are listed in an easily assessable format that makes sense. and can be assessed quickly. Price of the CD-Rom version is $50 per year. 1/4" Tapes are $122. Floppies (20 of them) are $199. 6-month updates included. This Data Base combines Trace Resource Books information with Able Data. For further information or to order call (608) 262-6966.

Equipment

Techniques, Devices, and Software

Single dimensional scanning techniques can be listener assisted or technology driven. They include:

Circular: Linear scan in a clockwise arrangement. Requires use of all eye muscles, a difficult visual/motor task. Pointers (which are always visible) may make the task cognitively easier that lights (which blink and disappear).

Step/Element: Linear scan. User activates a switch to advance cursor/light/pointer toward target. Time "is a switch." When user pauses, location is accepted as desired selection, or a second switch may be used.

Regular/automatic: User activates switch to begin linear (vertical or horizontal) scan and again to select item. Cognitively more difficult than circular because light/cursor disappears at end of row or column and then reappears.

Inverse: Linear scan. User holds down switch until cursor gets to desired item. Then, releases the switch to select item.

To enhance rate and provide access to more vocabulary, multi-dimensional approaches are used. These are more complicated; errors occur more frequently. High and light tech options are possible.

Row/column: Most commonly used technique in communication devices. User activates switch, device scans down rows until user activates switch, then device scans across columns, until user activates switch to select desired item.

Block/item: Best example is alphabet configurations on the Adaptive Firmware Card. Linear scan. When cursor arrives at block containing item, user activates switch. Then, cursor scans items until user activates switch to make choice.

Directed Scanning is a combination of direct selection and scanning. It generally involves multiple switches, i.e., technology. Those interviewed feel we don't pay enough attention to directed scan options in augmentative communication.

Directed scan: User has multiple switches (e.g., joystick) controlling the direction of the cursor. Items are selected when switch is released (i.e., inverse scanning.)
Note: Use of the mouse/track ball is a kind of free form directed scan. Proportional control devices are not addressed here.

Rate enhancement features are required if scanning techniques are to meet communication needs, particularly conversation. Some options employed are:

2-speed linear scan: (1-2 switches.) Regular or inverse. Scan proceeds rapidly to the vicinity of the target. User activates switch. Cursor proceeds more slowly until selection is made.

Letter/word prediction: Based on frequency or recency of occurrence, machine guesses letters/words as user creates message. A dynamic technique (i.e., user must constantly monitor the display and make decisions.) Interferes with automaticity. To date, evidence is limited re: performance effects. Woltosz reports impressive rates on a copying task.[1]

Encoding: User accesses vocabulary and linguistic structures through codes. e.g., Minspeak, abbreviation expansion, etc. Because codes often must be memorized, automaticity is affected. To what extent this affects scanning rates over time is unknown.

Software

Many programs are available to teach scanning techniques.[12] Programs listed in Table IV provide opportunities for individuals to practice various types of visual scanning techniques on the Apple II family of computers, and, at the same time, have fun! Remember to be aware of what you are teaching by doing a task analysis before you use a program.

Table IV. Scanning Training Programs

Don Johnston Devel. Equip. Wauconda, IL (312) 526-2683
- Interaction games $65 Linear and row/column scans and switch training. 2 persons.
- Learn to Scan $60. Horizontal, vertical scans. 8 programs with varying difficulties.
- Motor Training Games $35 14 games covering a wide range of scanning techniques.
- Make It and Make It in Time $60 each. Cause/effect, timing, judgement in time sequences, scanning techniques include group item and linear scans. Children and adults.
- Rabbit Scanner & Run Rabbit Run $29.95/$39.95 Horizontal scanning; timing

R.J. Cooper & Assoc, Dana Point. CA. (714)240-1912
- Early and Advanced Switch Games $30. Switch training.
- Switch progressions $37 More switch training, timing.

Computerade Products, Cardiff, CA 92007 (619) 942-3343
- Catch the Cow $13. Scanning trainer.
- Catch-and-Match $29.95. Follow-on to other training program. Also, a 2 to 20 choice array.

UCLA Intervention Prog. L.A., CA (213) 825-4821
- Where is Puff, Let's go shopping, Pic-Talk $18 each. Provide switch and scanning opportunities

Perhaps the most powerful tool we have is the Adaptive Firmware Card. With the Apple II gs computers, its flexibility provides access to a wide range of software. The technical manual is *(cont. page 6)*

University &Research

Research in Sverige

This issue highlights research activities in Sverige (Sweden) . . . the beautiful Nordic land where assistive devices are FREE!* Sweden will host the Fourth International Conference on Augmentative and Alternative Communication (ISAAC) August 12-18, 1990. Figure 1 illustrates the location of research projects** mentioned below. *Note: you may find this a useful guide if you travel to Sweden.*

In BROMMA, the **Swedish Institute for the Handicapped,** *(P.O. Box 303; Bromma, Sweden, S-151 26)* promotes and coordinates research and development activities in assistive technology areas. The institute also evaluates and tests products, disseminates information, conducts research, and builds the knowledge base in Sweden.

Projects include: Communication support for persons with speech handicaps (E. Olsson) - interventions for those with aphasia, dysarthria, laryngectomy, etc.

Come Again (M. Lundman & M. Magnusson) - use of computer conferencing systems by communication-impaired adults.

Inventory of communication aids for the mentally retarded and Communication for a profoundly mentally retarded adult (J. Brodin & L. Larsson) - use of technical aids by individuals with severe retardation.

Evaluation of computer-based aids and software (I. Friman) - recommends products for approval and provides information.

*Note: There is an assistive device center network that distributes approved assistive devices/aids: 90 centers for hearing impairments; 30 centers for visual impairments; 35 centers for orthopedic problems; and 35 for mobility/communication impairments. Five-seven Regional Resource Centers are being established to work in close affiliation with existing centers. They will provide direct services to multihandicapped clients, and disseminate information, education, and technical assistance to professionals.
**Information used is from Registration of Nordic Projects relating to disability (1988): Nordic Committee on Disability.

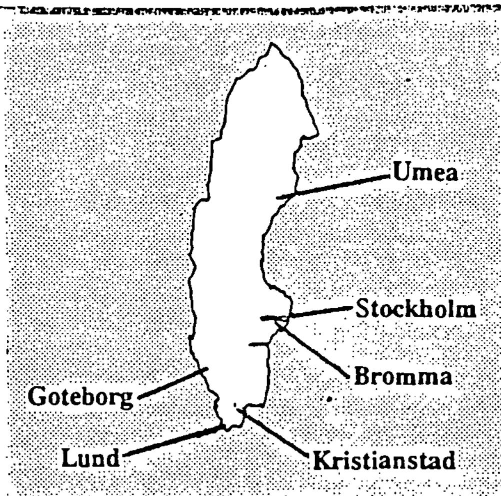


Figure 1. Sweden

In STOCKHOLM, **The Royal Institute of Technology** has always focused on technology applications for individuals with disabilities. AAC related projects within **The Dept of Speech Communication and Music Acoustics** *(Box 70014, Stockholm, Sweden S-100 44)*, are: Speech synthesis for disabled persons (R. Carlson and B. Granstrom) - Text-to-speech and Bliss-to-speech systems in multiple languages. A current focus is development of different speech styles and speaker types (i.e., child) and additional languages.

Development of speech output for communication and education and The use of computer and synthetic speech in teaching reading and writing (K. Galyas) - Applications of Multi-Talk and development of software to assist in teaching reading and writing, and in training those with aphasia, etc.

Word prediction for mobility and/or language impaired (S. Hunnicutt) - Software to assist individuals unable to read/write to construct written messages. Users provide limited information (e.g., 1st letter of words, no. of syllables.) A smart, natural language processing program in the computer "guesses" the word from a list of 10,000.

Also in Stockholm is **Stiftelsen ALA** *(Box 5410 Stockholm, Sweden, S-114 84)*. Projects which investigate behaviors affecting and interventions aimed at improving interaction include: Investigating the relationship among motor abilities, cognitive development and communicative ability in severely multiply handicapped (M. Granlund & C. Olsson) and Social interaction in special classes with mentally retarded (K. Gooransson).

Additional projects are aimed at developing teaching materials and methods, for specific populations. These are coordinated by the **Swedish National Institute of Teaching Materials** *(Box 27052 Stockholm, Sweden, S-102 51)*, and are carried out throughout Sweden

- Deaf-blind children and adults. L. Forsfalt. (KRISTIANSTAD).
- Multiply handicapped students using technical aids. G. Persson. (UMEA).
- Video & interactive video for mentally retarded. M. Liden (UMEA).
- Computers in education of handicapped pupils. U. Garthelson & H. Hammarlun (STOCKHOLM/BROMMA).

In LUND, the **Center for Rehabilitation Engineering (CERTEC), Lund Institute of Science and Technology** *(Box 118, Lund, Sweden, S-221 00)* has projects related to AAC (e.g., A hygienic suck-and-blow switch (I. Jonsson & L. Holmberg). They recently created a multidisciplinary training program at the Institute. Goals are to help professionals specialize in the assistive technologies area and develop related research skills.

In GOTEBORG, the University of Goteborg carries out research in the **Department of Education** *(Box 1010, Molndal, Sweden, S-431 26,)* and the **Department of Handicap Research** *(Brunnsgatan 2, Goteborg, Sweden, S-413 12)*. An example is: Communication in young children with a physical handicap and speech impairment (E. Bjorch-Akesson). A longitudinal interaction study of 7 preschool children and parents. Also, at the **Statens Institute for Laromedel** *(RPH-RH, Box 21071, Goteborg, Sweden, S-400 71)* the Software development & methods for children with severe motor problems project (E. Landin, E. Olsson) is ongoing.

Thanks to Lars Augustsson, Gunnar Fagerberg, Sheri Hunnicutt, Elisabet Olsson for sharing information about Sweden at the recent RESNA conference in New Orleans. Many thanks also to Karoly Galyas.

References

1 Woltosz, W. (1988) When is scanning or encoding better than direct selection? Presented at ISAAC Conference: Anaheim.

2 Fitts, P. (1965). "Factors in complex skills training." In R. Glaser (Ed.) Training research and education. Univ. of Pgh. Press.

3 Robb, M. (1972). The dynamics of motor-skill acquisition. Prentice-Hall, Inc.: Englewood Cliffs, NJ.

4 Robb, p. 65, see above.

5 Ratcliff, A. (1987). A comparison of two message selection techniques used in AC systems by normal children with differing cognitive styles. Unpublished dissertation, Univ. of Wisconsin-Madison.

6 Fried-Oken, M. (1989) Sentence recognition for auditory and visual scanning techniques in electronic augmentative communication devices. Paper presented at RESNA Conference, New Orleans, LA

7 Single input control assessment.$70 Easter Seal Communication Institute, 24 Ferrand Drive. Don Mills. Ontario, Canada M3C 3N2 (416) 421-8377.
Switch assessment program. $40. and Text Entry assessment program $40. Assistive Device Center. CSUS, 6000 J. Street Sacramento 95819 (916) 278-6442.

8 Goossens', C. & Crain, S. (in preparation) Utilizing switch interface with children who are physically challenged: Communication strategies. Available in early 1990, College Hill Press.

9 Light, J. (manuscript in preparation). Teaching automatic linear scanning for computer access to a severely physically disabled preschooler.

10 Bristow, D. (July, 1989). Personal communication.

11 VanTatenhove, G. (1988). Assessing the ability of very young & low-functioning students to use switches as a tool for controlling communication devices. In S. Blackstone, C. Cassatt-James, D. Bruskin (Eds.) Augmentative Communication: Implementation Strategies. ASHA:Rockville, MD.

12 Musselwhite, C. (1988) Using scanning switches with the microcomputer. Communication Outlook. 10:1, 12-13.

HINTS

1. The portable DECtalk is now available. It weighs 9 pounds, is battery operated, and is compatible with portable communication aids and personal computers with an RS232C port. Price, $1585. For further information Institute on Applied Rehab. Technology, The Children's Hospital, Fegan Plaza, 300 Longwood Avenue, Boston, Ma 02115

2. ACN does not list upcoming events. I'm making an exception because the topic is so pertinent to this issue. A conference on *Adaptive Play and Microcomputers* will be held in Asheville, NC, September 21-22, 1989. Musselwhite will share her creative approaches & expertise in a fun and energetic fashion. For information contact Irene Wortham Center, P.O. Box 5655, Asheville, NC 28813

13 Burkhart, L. has 3 excellent books full of ideas and instructions for making homemade technology and applying it creatively to enhance learning, 8503 Rhode Island Avenue, College Park, MD 20740.

14 Lahm, E. (1987). Software designed to teach young multiply handicapped children to use the computer for controlling their environment: A validation study. Unpublished doctoral dissertation: George Mason University, Fairfax, VA.

15 Cook, A., Hussey, S., & Murphy, J. (1988). Using technology in a diagnostic-therapeutic paradigm for severely disabled clients. Paper presented at ISSAC Conference: Anaheim.

16 Collier, B., Blackstein-Adler, S. Thomas, D. (1988). Visual functional issues in AAC-Clinical observations and implications. ISAAC Conference: Anaheim.

Resources

(Call these people about visual scanning. They have given it lots of thought!)

Jennifer Angelo, University at Buffalo, NY (716) 831-3141.

Peggy Barker, Stanford Rehab.Eng. Center, CA (415) 853-3345.

Roxanne Butterfield. Don Johnson Developmental Equipment, IL (312) 526-2682.

Al Cook, Assistive Device Center, CA (916) 278-6442.

John Costello & Maggie Sauer, Communication Enhancement Center, MA (617) 735-8392.

Cynthia Cress. Trace Research & Development Center, WI (608) 262-6966.

Carol Goossens', AC Service, U.C.P. of Greater Birmingham, AL (205) 251-0165.

Ina Kirstein. Oakland Public Schools, Pontiac, MI (313) 858-1901.

Robert Koch. 641 So. Palisade Drive, Orem, UT (801) 226-7997.

Arlene Kraat, Queens College, NY (718) 520-7358.

Janice Light, Hugh MacMillan Center, Ontario, CANADA (416) 425-6220.

Judy McDonald & Paul Schwedja. Adaptive Peripherals, Inc., WA (206) 633-2610.

Michael Palin, ACS Distributor, CA (707) 864-6321.

Sue Procter, Private practice, CA (408) 336-8265.

Ann Ratcliff. Loma Linda University, CA (714) 824-4998.

Gail VanTatenhove, Private practice, FL (407) 876-3423.

Walt Woltosz, Words+, CA (805) 949-8331.

Larry Weiss, Zygo Industries, OR (503) 684-6006.

Christine Wright, Stanford Rehab. Eng. Center, CA (415) 853-3345.

Augmentative
Communication
News

One Surf Way
Suite #215
Monterey, CA 93940

SALINAS, CA PM JUL 26 1989 93901

SALINAS RODEO ... WEEK ... JULY
Yosemite 25

RECEIVED
JUL 31 1989

PLEASE FORWARD

9

29 Exp-1/90
Carrie Brown
ARC/US Bioengineering
2501 Avenue J
Arlington TX 76006

Augmentative Communication News

March, 1990 Vol.3, No.2

INSIDE THIS ISSUE . . .

UPFRONT

As we settle into the 1990s, language and cognitive issues are being revisited *(e.g., literacy, language learning/relearning and the cognitive load of using augmentative techniques.)* This issue is about graphic symbols. It is also about rethinking some basic assumptions.

Many, many thanks to the professionals I interviewed *(see list of resources and references)*. Their insights, willingness to engage in long discussions and to share their research and experiences were invaluable! There are many unanswered questions about the use of graphic symbols in AAC. However, an army of researchers *(small, but powerful)* will help us find some answers. In the meantime, we'll need to be cautious as we make clinical decisions for and with people whose method of representing language is through graphics.

In **For Consumers**, research findings are shared and major issues are raised. The **Equipment** section highlights the need for professionals to be familiar with their "tools" (i.e., commercially available graphic symbol sets and systems). You'll also read about some new developments. In **Clinical News** the processes clinicians face in decision-making are discussed. For the **Governmental** section we shift gears and focus on *money*. **University & Research** highlights exciting work in the Netherlands at the

(continued on page 2)

For Consumers

"The medium is the message"

We are besieged by graphics. Graphics (pictures, signs, charts, logos, computer programs) inform, educate, influence and entertain us daily. Whereas artists, photographers, and advertisers use graphic symbols as a medium for expression, individuals with severe expressive communication disorders use graphic symbol sets and systems in qualitatively and quantitatively different ways, i.e., as a primary means for language representation and interaction. Graphic symbols may:

- 1. Stand for an individual's internal representation of the environment
- 2. Help individuals refine their knowledge of the world and influence their development of communication competence, cognition, language, and use of technology
- 3. Serve as medium of expression
- 4. Provide a window through which families/teachers/clinicians can learn about how individuals who don't speak, organize their experiences conceptually and linguistically.

As a field, we face major, unanswered questions about the processes connected with the perception, comprehension and use of photos, pictures, and commercially available graphic symbol sets and systems. Many feel it is time to refocus on these issues. After all, it is one's knowledge about the world and ability to express it that allows communication to occur.

Studies reveal visual perceptual skills develop early, but mature use of graphic symbols is not observed until ages 7 or 8 years when children use them to read, write, do math, etc.

- At 3 months babies can discriminate their mother's face from a stranger and between 2 strangers.
- By 6 months infants can "transfer" from live faces to highly realistic representations (color photographs) and "prefer" happy to angry or neutral faces.

(continued on page 2)

Augmentative
Communication
News

Upfront (cont.)

Institute for Rehabilitation Research (IRV).

Spring and/or Fall will soon be here (depending on your hemisphere)! I hope you'll take time to smell the flowers! Remember, The **Hotline Number** is (408) 649-3050.

Sarah Blackstone, Ph.D, Author, ACN

Table I. Studies of AAC symbols with normal children and adults

Author	Subjects	Symbols	Task	Result
Yovetich & Young (1988)	Adults	Blissymbols	Guess meaning of symbols for concrete concepts	Degree to which symbol represent concept affects learning
Fuller & Lloyd(1987)	Adults	Blissymbols	Rate according to complexity	Best predictors are: # semantic elements(comprehension) # of strokes (perceptual)
Luftlg & Bersani(1985)	Adults	Blissymbols		Translucency better predictor of learning than component complexity.
Fuller (1988)	Adults/ Children	Blissymbols	Taught to label or point to symbols	Adults learned more. Complexity influenced children's learning in positive way. Adult data generalized to nondisabled children. Learned superimposed symbols easily
Musselwhite	Children	Blissymbols	Recognition of symbols	Performance improves with age
Ruscello (1984)	Adults	Picsyms		Picsyms & Rebus more guessable.
Raghavendra & Fristoe (1990)	Children	Blissymbols standard/ enhanced	Guess meaning 10 actions, 10 attributes 10 entities	Identified perceptual features, letters shapes, numbers. Related to objects re: experience & world knowledge Focused on parts. Beginning reading
Bloomberg (1984)	Adults	Bliss, Picsyms PCS, PIC, Rebus	Rated translucency nouns, verbs, modifiers	Rebus most translucent across 3 linguistic classes
Ecklund & Reichle (1987)	Children	Blissymbols Rebus	Recall 15 symbols after repeated exposures	Performed well. Rebus recalled somewhat better. Children who pointed remembered more than those who used verbal labels
LePage Mills(submitted)	Children	PCS	Used during reading instruction	Positive effect on attitudes toward reading

Table II. Studies of AAC symbols with special populations

Author	Subjects	Findings
Mizuko & Reichle (1989).	Adults SMR	For nouns, Picsyms. PCS more readily associated with spoken words, than Bliss. For verbs and descriptors, no significant differences
Mirenda & & Locke (1989)	Children MMR-SMR some PH	Compared transparency of 11 symbol types representing objects. Results suggested a hierarchy of difficulty
Hurlbut, Iwata, & Green (1982)	Severe physical/ cognitive handicaps	Taught symbols representing known objects. Labels for iconic line drawings acquired more rapidly, maintained longer, generalized more frequently than labels for Blissymbols
Sevcik & Romski (1986)	Children Adoles. SMR	Learned arbitrary symbols as well as iconic symbols Those with nonfunctional language had more difficulty with line drawings than objects or photos.
Dixon (1981)	Adoles. SMR	Those who failed to match objects with photo may have had problems with perceptual properties. Using cut-out figures facilitated matching
Mineo (in prep)	Adults SMR	Match to sample paradigm within and between picture representation. Error analysis suggests size is a crucial feature.
Helm-Estabrooks Fitzpatrick Barresi (1982)	Adults w/aphasia	Description of Visual Action Therapy (e.g., progression from tracing, picture, matching, picture commands, gestrues) resulted in significant improvement
Esser & Mizuko(1989)	Children MMR	Use of graphic mode via VOIS 136 increased when aided language-stimulation was provided
Pecyna (1988)	Child w/ Downs	Developmental level and level of communicative function may influence effectivness of instructional program to teach graphic symbols

For Consumers (cont. from pg. 1)

- By 12-18 months infants demonstrate comprehension of photos/pictures paired with spoken words.
- 24 month old infants discriminate between real and pictured objects. Only after 28 months, can they use pictorial representations functionally.
- Between 2 and 7 years children make perceptual judgments about how things look, focusing on 1 variable at a time.
- 3 to 6 year old children can imitate a live model significantly better than a photo, a doll, or a line drawing. However, many 3 year olds have difficulty labeling actions of line drawings. (Mineo, review paper)

Research within AAC is addressing many issues directly related to graphic symbols. Table I summarizes some studies involving nondisabled adults and children as subjects. Questions include how easily symbols from various sets/systems are guessed, what perceptual and linguistic features may contribute to iconicity, and how adults and children approach graphic symbols. Table II, on the other hand, summarizes studies carried out with individuals who use, or are potential users, of AAC symbols to communicate. These studies begin asking what learning processes are involved for targeted groups to comprehend and use graphic symbols.

Issues related to graphic symbols are complex in AAC, but we've clearly made a start. For example, we have learned that:

1. Perceptual features can affect an individual's ability to attach meaning to symbols. Color, shape, complexity, figure/ ground, and size are important considerations.

2. While visual perception/discrimination influences symbol learning, it is an individual's ability to relate perceptually-based information to their experiences and knowledge of things, ideas, relationships or events that is most important.

3. Translucency seems to have a positive effect on learning when individuals are learning to pair an already known, spoken word with a symbol.

4. Iconic symbols (i.e., those easily guessed) are "easier to learn." However, most "iconic" symbols represent concrete objects/entities. Depicting actions, location, attributes, more abstract nouns, etc. requires the use of more abstract graphic symbols.

5. We need to take into account the learning styles of individuals. Different models for introducing and using graphic symbols with different populations are likely to emerge. For example, teaching symbols to normal talking individuals is different than

(continued on page 3)

AUGMENTATIVE COMMUNICATION NEWS is an independent, non-membership publication.
Author, Sarah W. Blackstone, Ph.D.; Publisher, Gary K. Poock, Ph.D.
Published by Sunset Enterprises, One Surf Way, Suite #215, Monterey, CA 93940
Subscriptions: By personal check U.S.$37 (1 yr), U.S.$67 (2 yrs), U.S.$97 (3 yrs). Single issues $8.
Overseas - U.S. $47 (1 yr.), U.S. $85 (2 yrs.), or U.S. $123 (3 yrs.).
Institutions, libraries, schools, hospitals, all others: Add $20 per year. Single issues $10.
Special rates for consumers and full-time students now available.
ISSN #0897-9278 Telephone (408) 649-3050

teaching symbols to nonspeaking individuals who already know what a word means, and is different than *teaching symbols to nonspeaking, cognitively delayed individuals with no language.*

6. Individuals with severe retardation can attach meaning to and use abstract graphic symbols after appropriate instruction.

7. It is still unclear if children and adults attach meaning to symbols using similar strategies.

8. Graphic symbols make language and concept development easier for some persons (e.g., those with autism, severe cognitive disabilities, and/or language disabilities).

9. Graphics symbols provide a bridge to literacy.

Major questions remain unexplored:

- What effects do graphic symbols have on cognition, language, communication, and literacy skills? How do we decide to use a graphic symbol set or system and then, which one?
- Does the early introduction of graphic symbols make a difference? How early? For which populations? Under what circumstances?
- What are the processes used by individuals with various handicapping conditions *(e.g., aphasia, dementia, severe spastic quadriplegia)*. What is the interaction between the individual's level of functioning, what they're exposed to, what they learn, and how they use graphic symbols?
- What instructional techniques should we use to enhance an individual's ability to perceive, comprehend, & use graphic symbols?
- When and how should we make transitions from one type of symbolic representation to another?
- What is involved in symbol learning beyond the initial phase? What about the 100th symbol, 200th symbol, etc.?

Equipment

What's happening and What's coming!

Commercially available graphic symbols were developed for multiple reasons, i.e., international peace, travel, to teach reading, and for use with severely speech-impaired individuals. Educators and/or speech-langauge pathologists have been involved in the development of nearly all AAC symbols. We owe our colleagues a hearty "thanks!" for the hours of work they continue to save us. Despite the increasing number of commercially available graphic symbols, most professionals are familiar with only a few...and that is what their clients use. Perhaps that's okay. Probably, it's not. We need to find out! Comprehensive, easy to read, thoughtful discussions by Lloyd and Vanderheiden (1986) and Musselwhite and St. Louis (1988) are a good place to start.

Graphic symbol sets[1] and systems[2] can be categorized according to their type of symbolic representation as McNaughton does in A Beginning Look at Graphics.[3]

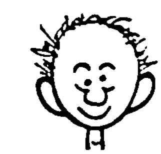
- **Letter-sound relationships:** e.g., Rebus
- **Direct representation, pictures and line drawings:** e.g., Compic, Oakland Picture Dictionary, Picture Communication Symbols, Imaginart symbols, Talking Pictures, PIC, and many more.
- **Logical, meaning-based depiction of meaning:** e.g., Picsyms, Blissymbols, Jet Era Glyphs
- **Portrayal of body movement, hand shape and or hand position of a sign or gesture:** e.g., World sign; Sigsymbols
- **Associations of other related concepts:** Minspeak icons

To illustrate 3 of the most commonly used types of symbolic representation, I interviewed professionals involved in bringing graphic symbols to "market." Among other questions I asked was "What is your favorite symbol?" Their responses are the symbols pictured on pages 3 and 4. *Note: This is all in fun,* and *a means of introducing them to you!*

1. Direct representation sets

Picture Communication Symbols

SILLY

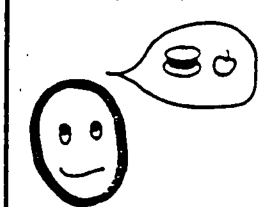


Roxanna Johnston developed these for children with mental retardation and autism as a simple, generic, iconic picture set. Current users are adults and children of all disability types. Some adults may find symbols childlike. 1700 + symbols (2 sizes) represent vocabulary that enables targeted populations to carry on everyday conversations. Materials: *2 dictionaries of symbols ($45-49); printed on colored or white paper. Stamps and other supportive instructional materials available. Computer programs generate symbols on the Macintosh using Hypercard (Boardmaker $299 and Board Builder $149)*

Distributed by Don Johnston Developmental Equipment. P.O. Box 639, Wauconda, IL 60084 and Mayer-Johnson. P.O. Box AD, Solana Beach, CA 92075-0838

Oakland Schools Picture Dictionary

HUNGRY

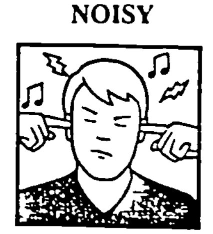


Ina Kirstein originally developed this set to provide students with moderate retardation a symbol set they could understand with little or no instruction. Current users include children and adults of all disability types. Not appropriate for some with low vision or individuals who can manipulate abstract concepts, symbols and syntax. Vocabulary of 500 symbols (3 sizes) selected to address needs of target population. Materials: *Picture Dictionary ($45).*

Distributed by Don Johnston Devel. Equip. and Oakland Schools, Communication Enhancement Center, 2100 Pontiac Lake Road, Pontiac, MI 48054

Imaginart

NOISY



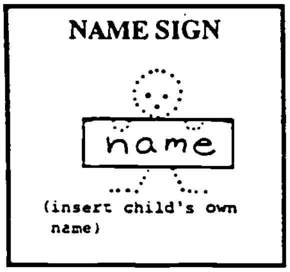
Cindy Drolet developed this initially for travelers and adapted it for use with children (cont. p. 4)

1 Set of symbols. No clearly defined expansion rules
2 Set of symbols designed to work together to allow maximum communication. Has rules for expansion beyond current system.
3 Available from the Easter Seals Communication Institute, 250 Ferrand Drive, Don Mills, Canada. Send $5 with order.

and those with mental retardation. Currently adults and children use them. Not appropriate for visually impaired and the very young or elderly. Vocabulary of 600 symbols (2 sizes) is oriented toward target population. Color and black and white symbols. Materials: *Pick 'n Stick Color Packs ($43.95), Touch n' Talk Communication Stickers ($28.95) and Touch 'n Talk Micros ($25.50).*
From Imaginart. P.O. Box 1868, Idyllwild, CA 92349 and Winslow Press outside U.S.

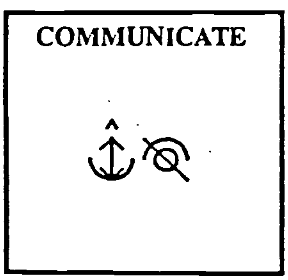
2. Logical, meaning-based systems

Picsyms



Faith Carlson developed the system as a dynamic therapy tool for preschool children. She teaches adults to draw (so symbols can be created at the time they are needed.) The system has a way to handle grammar. Used with all ages; not appropriate for people with low vision and adults who object to child-like nature of symbols. Vocabulary of 800 symbols from a range of categories. Materials: *Instruction book and dictionary ($25).*
Distributed by Don Johnston Developmental Equipment and Baggeboda Press, 107 N. Pine Street, Little Rock, AR 72205

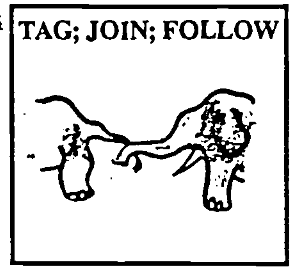
Blissymbols



Shirley McNaughton has been involved in bringing this pictographic, ideographic system composed of meaning-based units and some arbitrary shapes to many countries. It has a grammar, and is used by children and adults. Not appropriate for individuals who show no interest in classifying or in communication. Simplicity of shape and line allows for writing. Vocabulary of 3,000 symbols updated regularly by Blissymbolics Communication International. They also provide training. Materials: *Blissymbols for Use & Supplements $45; Picture Your Bliss $70. Also, texts, writing, and self-study materials.* *NOTE:* New programs for the Macintosh written in Hypercard will provide quick access to 2500 Blissymbols [Access Bliss, $250 Can.] and early independent reading experiences [Story Bliss, $100 Can.].
Ebsco Curriculum Materials, P.O. Box 1943, Birmingham, AL 35201 in U.S.;
Easter Seal Communication Institute, 250 Ferrand Drive, Suite 200, Don Mills, Ontario, Canada M3C 3P2 outside U.S.

3. Associations of related concepts

Minspeak Icons



Bruce Baker, (J. Bruno/G. Van Tatenhove)

Developed as examples to show the flexibility of the semantic compaction approach. The purpose is to provide a code so a user can get language out of a computer. Baker emphasizes icons are not meant to function as symbols on communication boards. Icons relate to the individual's ability to associate multiple meanings. Thus, it is immaterial whether listeners "understand" them. Icons are culturally bound and often perceptually complex.
Available with software for TouchTalker and LightTalker from Prentke Romich Co. 1022 Heyl Rd. Wooster, OH 44691

Graphic Technologies in the Making for AAC

1. Phonic Ear's VOIS Shapes, created by Howard Shane of Children's Hospital in Boston and Ronnie Wilbur of Purdue University, will be available by early summer. This software for the VOIS 160 draws upon the production characteristics of American Sign Language (ASL), yet users do not need to know ASL to generate thousands of words. Basically, the program encodes messages, using the components of sign: *location, hand shape, and movement.* Words are accessed from a single key pad comprised of sign language components. The user selects words making no more than 3 "hits" to access a word.
For information, contact Bill Forde or Yvonne Ho at Phonic Ear Co. 250 Camino Alto, Mill Valley, CA 94941 (800) 227-0735.

2. Words + , Inc: Talking Screen software for IBM compatible computers was just released.*($1195).* Users select symbols from up to 10 levels (using scanning, joystick, or mouse-emulating input). Hundreds of symbols may be accessed on changeable displays reducing cognitive demands for user and increasing access to vocabulary stored on different "levels." Spelling mode with abbreviation expansion also available. Up to 128 cells per level.
For information, contact Walt Woltoscz, P.O. Box 1229, Lancaster, CA 93584. (800) 869-8521

3. Bio-Engineering Department, Association for Retarded Citizens is developing an integrated communication and environmental control system for individuals with limited cognitive function. The prototype is controlled by voice recognition technology. Users can select a message or control their environment using photographic quality symbols on a customized, dynamic display.
For more information contact Maggie Sauer, 2501 Avenue J, Arlington, TX 76006. (817) 640-0204.

4. CRTL, University of Arkansas-Little Rock is developing a graphic-based, multi-purpose communication device using Hypercard. Key features are the dynamic display, digital speech which can be modified by a clinician to improve quality (e.g., provide inflection), the ease with which displays are created, and random access to digitized speech, graphics, text, and environmental controls. Automated record keeping tracks user's responses and may suggest ways to reconstruct displays to improve fluency.
For more information contact Alan Van Bierlet, 2801 South University, Little Rock, AK 72204 (501) 569-3423

The future role of graphics in AAC

Simulations and animation may reduce the cognitive load of graphic symbols and make transitions from very realistic photographs to more abstract representation easier by systematically removing detail. Perhaps the most exciting possibility is that computer graphics may make it possible for individuals to do what they need to do, i.e., generate their own symbols (Verburg, 1989).

Clinical News

"It is not on my board"

Symbols represent vocabulary[4], which in turn, represents the concepts/messages/knowledge individuals may convey. Despite our most valiant efforts the reality of "it's not on my board" resounds! It can mean many things: 1) the desired vocabulary/symbol truly is not there 2) the user can't find the appropriate symbol 3) the user does not know what the symbol means 4) the communication partner doesn't understand the symbol/message, and so on.

Multiple clinical decisions about graphic symbols are made: 1) Whether or not to use graphic symbols? 2) Which ones to use? 3) How to configure a symbol display for a communication board or overlay for an electronic communication device or computer, 4) How to develop concept/symbol pairings 5) How to facilitate the use of symbols to accomplish various communication tasks, and 6) How to evaluate what's happening?

1. Graphic symbols: Yes or no?

Note: The word "symbol" implies "The comprehension or use, inside or outside communication situations, of a relationship between a sign and its referent, such that the sign is treated as belonging to and/or substituting for its referent in a variety of contexts; at the same time the user is aware that the sign is separable from its referent, that it is not the same thing. (Bates, p. 43 in Savage-Rumbaugh & Rumbaugh, 1980)

Graphic symbols are vital to many nonspeaking people. If an individual can perceive features of two dimensional stimuli, graphic symbols can be considered for use in interactive communication and as a powerful learning tool.

2. Which symbol set or system?

Note: "In order to handle the world with maximum competence, it is necessary to consider the structure of things. It is necessary to become skilled in manipulating systems and abstract forms and patterns." (Donaldson, 1985, p. 82)

Professionals should be cautious and question their assumptions. For example, symbol hierarchies may purport to define a developmental progression of symbol acquisition (e.g., objects; color photos, black & white photos, pictures, line drawings, Blissymbolics, traditional orthography). However, no strong evidence exists; and individuals who use AAC do not necessarily conform. Deciding what type of symbols to use with an individual requires consideration of:

- The highest level of graphic symbols the individual can use
- The impact of today's decisions on an individual's future
- The amount of instruction required for the individual to attach meaning to the symbols, and
- Materials available to the clinician.

Important questions to consider are:

PERCEPTION: Given an individual's visual acuity and cognitive ability, what types of symbols are perceived? What perceptual features are important to the individual: size, color, line, figure/ground, shape, redundancy, realism, closure, spacing? Can the individual transfer meaning from 3-dimensional to 2-dimensional symbols? Is there a figure ground problem? Does cutting out the figure in a photo help? Does taping a background to a 3 dimensional referent help?

COMPREHENSION: The presence or degree of language skills should be known before making major decisions. Does individual attach meaning to symbols? which ones? under what circumstances? Consider time and effort required for learning. Consider potential short and long term benefits.

USE: Can individual use symbols to access synthetic speech, computer programs, and written text? Does the individual know how to use pictures/symbols to direct actions of others? Does individual use graphic symbols in conjunction with other forms of expression (gesture, facial expression)? Or, do they "stop communicating" and "start pointing" when a communication display is introduced.

Those interviewed had additional comments:

- Symbols can be drawn from more than 1 set/system to meet an individual's needs. This should be done carefully.
- Whether clinicians should use the same symbol set across individuals (e.g., in a classroom) is controversial. Some feel it is advantageous because it provides a common "language." Others disagree indicating symbols should be selected on the basis of each individual's current and future needs. They suggest using a gloss *(a written word or a picture above/below the user's symbol)* to insure communication partners understand symbols.
- Transitions from one type of symbol set/system to another require careful planning. Most caution that teaching up or down a "hierarchy" is ill advised *(i.e., don't start with objects, then go to photographs, then pictographs, etc.)*. NEVER take away what exists when introducing something new. Allow for a gradual transition after a period of instruction.

3. What about symbol displays?

After specific symbols representing needed vocabulary are selected, they can be displayed on an overlay or board so the individual can select them, as needed. The purpose of communication displays will change over time. See Table III for examples of changing goals.

Table III. Communication displays
(adapted with permission from McNaughton (1989). A Beginning Look at Graphics.)

INITIAL DISPLAYS
- teaches and reinforces the value of communication
- introduces the individual's beginning vocabulary
- presents the individual's beginning vocabulary
- allows for rapid vocabulary growth
- offers initial structure to the vocabulary
- prioritizes needs

TRANSITIONAL DISPLAYS
- provides opportunities to evaluate appropriateness of initial representation system and display organization
- accommodates growing vocabulary
- allows restructuring vocabulary based on experience
- provides AAC user with opportunities to assume more decision-making with regard to vocabulary and its organization

ADVANCED DISPLAYS
- should be planned and directed by AAC user as much as his/her capabilities allow
- should always grow and change
- should be integrated into the individual's total communication system, i.e., serve as a complement to synthetic speech, signing, vocalizations, etc.

Excellent descriptions are available to help clinicians design communication boards or overlays (e.g., Musselwhite & St. Louis(1988); Brandenburg & Vanderheiden (1989); McNaughton (1989). Considerations include:

- Size, shape of display, spacing between symbols, boldness of symbols
- Color of symbols. Color of background. Background color may assist an individual to locate the symbol, as well as facilitate the development of categorization and syntax.
- Other figure/ground considerations
- Relative size of symbols and words.
- Rate and accuracy of user's ability to make selections

(continued on page 6)

[4] Vocabulary selection issues were discussed in Vol. 1, #5 of ACN.

- Number of symbols user and partners can interpret on a display; Number of symbols that can be displayed on an electronic device/keyboard.
- Availability of instructions for partners
- Vocabulary arrangement/organization of symbols (categorical/linguistic, e.g., Fitzgerald key)
- Mechanism for adding vocabulary
- Location of core symbols on multiple displays/overlays.
- Whether to use a gloss (e.g., printed word or a picture) to insure partners can interpret symbols.
- Type of permanent displays (single display, multiple displays, combination) and use of more temporary or situational topic/mini-boards

4. & 5. Developing comprehension and use of symbols

Note: What is remembered in any situation depends on the physical and psychological context in which the event was experienced. the knowledge and skills the subject brings to the context, the situation in which the (person) was asked for evidence of remembering and the relation of what the (person) remembers to what the (situation requires). (Jenkins, 1974, p. 793)

The key to symbol learning is the instructional process. How symbols are taught influences how they are used and how listeners regard the communication acts of graphic symbol users. Asking an individual to "show me" "point to" "tell me" is not teaching, it is testing. Table IV is a partial list of responses I received to the question "what mistakes are we making in our clinical work?"

Table IV. Clinical Practices We Need to Change

- Testing before symbols are introduced.
- Giving individuals a communication board and "hoping" people use it. Not training listeners.
- Partners not using board.
- Not teaching in actual situations.
- Starting with too many symbols.
- Selecting symbols that do not have functional importance to individual.
- Failing to realize the importance of the cosmetic value of a display.
- Not knowing how to set up a display for fluency. Not organizing displays.

Instructional strategies for teaching the comprehension and use of symbols are based on good instructional techniques and principles, including:

- modeling the use of graphic symbols (aided language stimulation)
- providing repeated examples of the symbol paired with its referent
- highlighting salient features (e.g., using intonation or a flashlight to call attention to the symbol/referent pairing)
- providing the symbol as person's attention focuses on its referent
- providing instruction during *meaningful* activities, which might include the use of play, scripts, role playing, etc.
- providing multiple opportunities for individuals to use symbols, etc.

No matter what approaches are used, partner training is always necessary. In addition, we need to help individuals learn to "solve" their own communication problems by fostering their use of multi-modalities and encouraging their creativity.

6. Evaluation

Complete mastery of visual symbols can not be assumed without demonstration of their functional use. However, along the way, we need to be looking at comprehension and perception and the processes involved in learning. Individuals who may appear to be nonlearners may, in fact, be learners when we examine their comprehension. Although aspects of graphic symbol learning can be examined separately, they are interdependent. Whether or not an individual "masters" graphic symbols may ultimately depend on our ability, as professionals, to stop talking, deciding, and doing unto the people with whom we work long enough to "listen to them learn", i.e., watch what they do, listen to what they "say", and find out who they are and how they learn.

Governmental

Grants and Contracts
Soon to be announced in U.S.

New awards to U.S. states for Title 1 monies will be announced at the end of March. Congratulations, in advance, to the second round of winners!

More good news. Title 2 of the Technology Assistance Act of 1989 (P.L. 100-407) has been funded for 1990. Watch the Federal Register in March and April for a Notice of Proposed Rule Making (NPRM). Just about anyone is eligible to apply for Title 2 funds and awards will be made in the range of $125,000 per year. Funded projects may be for research and development of equipment, model service delivery programs, demonstration projects. Also, a contract will be announced soon in Commerce Business Daily for an agency/institution, etc. to conduct a feasibility study for a National Information and Referral Network for Technology.

Note: *It is probably a good time to reread Title 2 of P.L. 100-407.*

For additional information, Contact Carol Cohen, NIDRR, 400 Maryland Avenue, S.W., Washington, D.C. 20202-2645, (202) 732-5066.

Hints

Thanks to all for sharing ideas and information!!!!

CAROLINE MUSSELWHITE: **Switch Power Pad.** Lekotek of Georgia is adapting Power Pads to accept 9 switch inputs. Caroline reports it is "super" for programs such as UCLA's Wheels on the Bus. They've modified the Talking Power Pad with 17 nursery rhymes on 1 side and 10 games on the other. The switch power pad is available from Reach, Inc. 890 Hearthstone Drive, Stone Mountain, GA 30083.

KAREN CASEY: **Sheet velcro** (e.g., velfoam). 36" x 54" sheets (in a variety of colors blue, pink, yellow, and lavender) are available at approximately $16 per yard. She says it works much better than strips of velcro sometimes. Karen gets it from DMC Sales, Inc. 2015 West C street, Kannapolic, NC 28081. Ask your local folk!

INA KIRSTEIN: In constructing communication boards/displays, try using **transparent, self adhesive, colored "film."** It's like contact paper. Brand names include Chart Pak, Form-X-Film, and Zip-A-Tone. Use it to provide background colors for Fitzgerald key or color-code miniboards, etc. You can put it on top of display or use it as a background with white/black symbols on top.

CAROL GOOSSENS: To take **uncluttered photos**, put object on a table in front of window. Put a piece of poster board behind the object. Make sure the figure/ground (object/poster board) offers good contrast. Try bright yellow poster board.

University & Research

Institute for Rehabilitation Research, The Netherlands

Major research is underway at the Institute for Rehabilitation (IRV) in Hoensbroek. IRV was founded in 1981 through the efforts of *The Netherlands Organization for Applied Scientific Research* (TNO) in The Hague, the *University of Limburg* in Maastricht, and the *Lucas Foundation for Rehabilitation* in Hoensbroek. IRV's mission is to conduct multidisciplinary, rehabilitation research in 4 major areas:

1) **Communication** (Coordinator, Hans van Balkom). Investigation of variables related to interaction and the use of technical aids and nontechnical AAC components (e.g., graphic symbols, manual signs).

2) **Mobility and transportation** (Coordinator. Hok Kwee). Wheelchair design and car adaptations, e.g., transfer from wheelchair to car.

3) **Patient-related research** (Coordinator, Joop Schuerman). Methodologies for measuring the outcome and/or processes of treatment.

4) **Independent living and working** (Coordinator, Frank Vlaskamp) Methods for defining needs and effectiveness of clinical treatment on home and work situations. Development of technologies that promote independence at home and work using robotics and electronics.

The IRV, directed by M. Soede, employs approximately 50 professionals and maintains financial and collaborative partnerships with its founding agencies and several government agencies: *Departments of Health Care, Education and Research, Social Affairs, and Provincial.* Private companies/groups also support special projects.

IRV collaborations in augmentative communication (AC) extend internationally:

Belgium: Catholic University,
Canada: Hugh MacMillan Medical Centre*,
France: The Technical University
United Kingdom: Microcomputer Center, University of Dundee*
United States: A.I. Dupont Institute*; Purdue University*

AAC Research Program

Under the direction of Hans van Balkom, researchers from multiple disciplines (linguistics, phonetics, rehabilitation engineering, ergonomics, psycholinguistics, artificial intelligence, speech therapy and occupational therapy) are contributing to 7 major focus areas:

1. Communication interaction *(Project director, Hans van Balkom)* The goal is to develop a clinically applicable linguistic method for assessment of communication skills in children with a) specific language development problems, b) their caregivers. Twelve toddlers with specific language disorders and 6 with normal development were followed for eighteen months. Videotaped free-play sessions with caregivers were transcribed and analyzed using the Interaction Analysis of Communicative Abilities (IACV). Computer assisted interaction analyses programs and a Handbook for IACV will be available in Fall, 1990. Note: The IACV is being adjusted for use with other clinical groups.

2. Rate enhancement using efficient input devices *(Project directors, Harry Kamphuis & Jan Duimel)* Two techniques to enhance the communication rates of individuals generating text are being evaluated. The systems employ different rate enhancement strategies, based on word and word part frequency counts (COMBI) or quadgram tables (KATDAS). Frequency counts are based on written and spoken Dutch, linguistic rules, ergonomic knowledge about keyboard construction, feedback, and information processing. Major issues are 1) using effective feedback to control/correct devices and 2) reducing mental load and typing errors. Prototypes of KATDAS and COMBI will be available in late 1990.

3. Clinical assessment for AC *(Project director, Eelke Oostinjen)* This project has established AC assessment teams in rehabilitation centers and stimulated the foundation of the Dutch Information and Counselling Center for AC *(CIAC)*, a national information dissemination resource centre in Huizen.

4. Graphic symbol communication *(Project director, Marguerite Welle Donker-Gimbrere)* This project is conducting surveys to identify graphic system use and will describe characteristics of symbols in use. An empirical study of symbol characteristics to determine their useability in relation to user abilities is planned. Goals are to assist clinicians and teachers to improve how they select and train the use of graphic symbols.
Purdue University and Indiana State University are cooperating in this project.

5. Motor assessment for selection of input devices *(Project director, Jan Goezinne)*. Movement amplitude, visual feedback, and movement direction are used in tasks to assess motor capabilities for accessing AC devices. Products include: several input devices for children with motor impairment and 2 computer-assisted motor assessment techniques.

6. Speech output and AC devices *(Project director, Eelke Oostinjen)*. This project investigates features to consider when developing and constructing speech output communication devices and will result in the development and evaluation of a flexible speech-output device for Dutch speakers.

7. Telecommunication adaptations for AC users *(Project directors Jos van Well & Thijs Soede)* This European Community (EC) project is developing a list of technical requirements to allow AC users access to a telecommunications network.

In addition to research activities, the IRV-AC program initiates and supports other endeavors in the Netherlands and Flanders:

- Initiation of ISAAC-NF in 1989 (190 members)
- Communicatie Drieluik, a quarterly journal. supported by ISAAC-NF, CIAC, and the Centre for Blissymbolics-NF
- 1990-91 Conferences in NF & Europe: Developmental Disabilities in Children (ISAAC-NF) in Belgium; ECART Conference in Maastricht; and Psycholinguistic & AC

Fellowships are available. For additional information & a list of related publications, contact Hans van Balkom.

Address: IRV, Zandbergsweg 111,
6432 CC Hoensbroek - The Netherlands
Telephone: 31 45 224300
Fax : 31 45 226370

* Programs featured in past issues of ACN

Augmentative
Communication
News

Resources and References

Bruce Baker, Semantic Compaction Systems, 801 McNeilly Road, Pittsburgh, PA 15226, (412) 563-6377 FAX (412) 563-3167

Marguerite Welle Donker-Gimbrere, IRV, Zanbergsweg 1112, 6432 CC Hoensbroek, NF

Faith Carlson, Rt. 1, Box 2315, Unity, ME 04988. (207) 437-2746.

Cindy Drolet, Imaginart Communication Products, P.O. Box 1868, 25680 Oakwood St., Idylwild, CA 92349 (714) 659-5905

Carol Goossens', UCP of Greater Birmingham, Birmingham, AL (205) 251-0165

Roxanna Johnson, Mayer-Johnson Co., P.O. Box 1579, Solana Beach, CA 92075-1579 (619) 481-2489

Ina Kirstein, Communication Enhancement Cntr, 2100 Pontiac Lake Rd., Pontiac, MI 48054 (313) 858-1901.

Janice Light, Dept. of Comm. Disorders, Penn State University, Moore Building, University Park, PA 16802 (814) 863-2013.

Lyle Lloyd, Dept. of Special Educ., Purdue University, W. Lafayette, IN 47907 (317) 494-7333.

Beth Mineo, A.I. Dupont Instit., Div. of Applied, Sci. & Eng., 1600 Rockland, Rd., Wilmington, DE 19899 (302-651-6836).

Shirley McNaughton, c/o Easter Seal Communication Institute, 250 Ferrand Drive, #200, Don Mills, Ontario, Canada M3C 3P2

Mary Ann Romski, Yerkes Primate Center, 954 Gatewood Rd. Atlanta, GA 30322 (404) 243-8287.

Hans Van Balkom, IRV, Zanbergsweg 1112, 6432 CC Hoensbroek, The Netherlands

Geb Verburg, Hugh MacMillan Medical Centre, 350 Rumsey Rd. Toronto, Canada M4G 1R8 (416) 424-3805.

Bloomberg, K. (1984). The comparative translucency of initial lexical items represented in five graphic systems. Unpublished master's thesis, Purdue University, IN

Brandenburg, S. & Vanderheiden, G. (1988). Communication board design and vocabulary selection. In L. Bernstein (Ed.) The vocally impaired: Clinical practice and research. Gruen & Stratton: Philadelphia.

Dixon, L. (1981) A functional analysis of photo-object matching skills of severely retarded adolescents, Journal of applied behavior analysis. 14:465-478.

Donaldson, M. (1978). Children's minds. Fontana Paperback: London. (p. 82)

Ecklund, S. & Reichle, J. (1987). A comparison of normal children's ability to recall symbols from two logographic systems. LSHSS. 18:, 1, 34-40.

Esser, J. & Mizuko, M. (1989). Intervention effects on AC system user's interaction skills. Presented at ASHA Convention: St Louis, MO.

Fuller, D. & Lloyd, L. (1987). A study of physical and semantic characteristics of a graphic symbol system as predictors of perceived complexity. AAC. 3:1, 26-35.

Fuller, D. (1988a,b). The associative learning of Blissymbols varying physical configuration by cognitively normal preschool children. and Further comments on iconicity. AAC, 4:3, 180-181.

Helm-Estabrooks, N., Fitzpatrick, P., Barresi, B. (1982) Visual action therapy for global aphasia. JSHD. 47: 385-389.

Hurlbut, B., Iwata, B. & Green, J. (1982). Nonvocal language acquisition in adolescents with severe physical disabilities: Blissymbols versus iconic stimulus formats. J.of Applied Behavior Analysis. 15, 241-259.

Jenkins, J. (1974). Remember that old theory of memory? Well, forget it! American Psychologist. November. 785-795.

Lloyd, L. & Vanderheiden, G. (1986) Communication systems and their components. In S. Blackstone (Ed.) Augmentative Communication: An Introduction. ASHA: Rockville, MD.

Luftig, R. & Bersani, H. (1985). An investigation of two variables influencing Blissymbol learnability with nonhandicapped adults. AAC. 1:1, 32-37.

McNaughton, S. (1989) A beginning look at graphics. Easter Seal Communication Institute: Toronto, Canada.

Mineo, B. (an unpublished review paper). Pictorial representation: An analysis of the abilities of two young adults with severe mental retardation.

Mineo, B. (in preparation). Visual symbol perception and understanding in infants and young children.

Mirenda, P. & Locke, P. (1989). A comparison of symbol transparency in nonspeaking persons with intellectual disabilities. JSHD. 54:131-140.

Mizuko, M. & Reichle, J. (1989) Transparency and recall of symbols among intellectually handicapped adults. JSHD. 54:627.633.

Musselwhite, C. & Ruscello, D. (1984) Transparency of three communication systems. JSHR. 27:436-443.

Musselwhite, C. & St. Louis, K. (1988). Communication programming for persons with severe handicaps. Little Brown.: Boston

Pecyna, P. (1988) Rebus symbol communication training with a severely handicapped preschool child: A case study. LSHSS. 19: 2, 128-143.

Raghavendra, P. & Fristoe, M. (1990). "A spinach with a V on it": What 3-year-olds see in standard and enhanced Blissymbols. JSHD. 55:1,149-159.

Reichle, J. & Yoder, D. (1985). Communication board use in severely handicapped learners. LSHSS. 16:3, 146-157.

Savage-Rumbaugh, E. & Rumbaugh, D. (1980) Requisites of symbolic communication--Or, are words for birds? The psychological record. 30:305-318.

Sevcik, R. & Romski, M.A. (1986). Representational matching skills of persons with severe retardation. AAC. 2:4, 160-164.

Sevcik, R. & Romski, M.A. (in preparation). AAC symbols: Their roles in communication acquisition for persons with severe cognitive disabilities.

Verburg, G. (1989) Closing the acquisition versus composition theme. Communicating Together. 5:3, 19-20.

Yovetich, W. & Young, T. (1988) The effects of representativeness and concreteness on the "guessability" of Blissymbols. AAC. 4:1, 35-39.

Augmentative
Communication
News

One Surf Way
Suite #215
Monterey, CA 93940

SALINAS CA
MAR 20
1990
93901

Yosemite

PLEASE FORWARD

29 Exp-1/91
Carrie Brown
ARC/US Bioengineering
2501 Avenue J
Arlington TX 76006

Publications Pertaining to the Sound-to-Speech Research Project

Brown, C. C. (1989). Research focusing on freedom of choice, communication, and independence using eyegaze and speech recognition assistive technology. In A. VanBiervliet & P. Parette (Eds.), Proceedings of the First South Central Technology Access Conference (pp. 27-34). Little Rock, AR: University of Arkansas at Little Rock.

The Arc (Producer), & Finley/Pierce & Associates (Director). (1991). Voices of the future [Videotape]. Arlington, TX: The Arc.

Brown, C., Sauer, M., Cavalier, A., & Wyatt, C. (1992). Sound-to-speech translation system utilizing graphics symbols. Proceedings of the RESNA International '92 Conference (pp. 447-449). Toronto, Canada: RESNA.

Brown, C., Wyatt, C., Sauer, M., & Cavalier, A. (1992). The sound-to-speech translation utilizing graphic symbols. Proceedings of the Seventh Annual Conference on Technology and Persons with Disabilities. Northridge, CA: California State University, Northridge.

Brown, C., Cavalier, A., Sauer, M., & Wyatt, C. (1992). The sound-to-speech translation system utilizing photographic-equality graphic symbols. Johns Hopkins APL Digest, 13(4), pp. 1-9.

Brown, C., Sauer, M., Cavalier, A., & Wyatt, C. (1992). Speech recognition and graphics research for persons having mental retardation. Augmentative and Alternative Communication, 8(2).

** Researchers on this project plan to submit this research to a professional referred journal with the net three months.

(1989). In Proceedings of the First South Central Technology Access Conference. Little Rock, Arkansas: University of Arkansas, 27-34.

Research Focusing on Freedom of Choice, Communication, and Independence Using Eyegaze and Speech Recognition Assistive Technology

Carrie Brown, Ph.D.

The Association for Retarded Citizens of the United States

Bioengineering Program

People with disabilities have realized numerous benefits through the application of contemporary technological advances. The Association for Retarded Citizens of the United States is committed to insuring that these benefits are enjoyed by this country's 6,000,000 children and adults who are mentally retarded. Recognizing not only that technology was finding its way into everyone's life, but also that it could be applied to special problems that had eluded satisfactory solutions, the ARC established its Bioengineering Program in 1982. The Bioengineering Program, as a research and development program, has three major goals: first, to investigate procedures and techniques that improve the use of existing technology by children and adults with mental retardation; second, to develop new assistive devices and systems to address unmet needs; and third, to provide technical assistance to parents and professionals.

Research Need

There are more than two million children and adults in the United States whose physical and/or mental impairments are so severe that they are unable to communicate with other persons or interact and learn from their environment in an effective manner (Bricker & Filler, 1985). Disabilities of this degree impact people with cerebral palsy, paralysis, spina bifida, mental retardation, and cardiovascular disease/stroke, and severely limit their ability for independent living, effective education, and productive employment. Individuals with severe physical impairments are frequently bed- or wheelchair-bound with very limited control over even gross motor movements. If it exists at all, oral communication is oftentimes limited at best, with the degree and combinations of disability impacting the severity of this problem. Individuals with severe disabilities are usually totally dependent on others to discern their basic needs such as thirst, hunger, or toileting, and to act on those needs. They also are prevented from exercising simple choices that are in line with their desires, such as turning on the TV or listening to music. Caregivers, family, therapists, and teachers struggle to find a pathway to assist these individuals in functionally effective ways. For lack of such a pathway, even personal computers with their unique educational strengths are powerless to facilitate their developmental progression. Often these individuals are denied by their handicaps and society's response (or lack of response) to them the social interaction, opportunities for education and productivity, and personal fulfillment to which everyone is entitled. Caregivers, teachers and therapists are also severely impacted by these multiple handicaps in that they are required to provide extensive and near constant attention (Fredericks, 1985).

Despite the serious need that has existed to compensate for the limitations and extend the abilities of individuals with severe physical impairments and possibly mental retardation, until recently these individuals have largely been ignored in the development of technology-based aids and related training procedures. Very few of these people possess sufficient motor control to use a computer keyboard, and current attention has almost exclusively focused on the use of physically-operated microswitches as alternate access modes (*e.g.*, Ellington, 1986; Esposito, 1985; Meehan, Mineo, & Lyon, 1985; Parette, Strother, & Hourcade, 1986; Wright & Nomura, 1985; Torner, 1986). In recent years, researchers have begun to address these serious needs in more creative ways with the cutting edge technologies of voice recognition and eyegaze/headpointing (Brown, 1989; Brown & Cavalier, 1989; Brown, Cavalier, & Tipton, 1986).

Bioengineering Speech Recognition Research

According to Raymond Kurzweil, it is recognized that speech recognition as a means of interfacing a user with handicaps "offers real hope for personal communication, education, and gainful employment. Voice is the most natural, effective form of communication" (Weintraub, 1987). Not only is voice recognition recognized in the field of disabilities for its potential, but in the business arena it is acknowledged that "when combined, natural language and voice recognition represent one of the most significant advances in the history of computers" and that by the mid 1990's voice input will be the rule rather than the exception for computer control (Lazzaro, 1986).

Since 1984, staff from Bioengineering Program have been developing a research track called the Freedom of Choice and Independence which utilizes speech/voice recognition technology or eyegaze/headpointing technology for freedom of choice, communication and environmental control. Brown (1985) and her colleagues have sought to improve access to the power of the microcomputer in extending control over one's immediate environment and to communicate in a normalized fashion to people with severe/profound mental retardation and severe physical impairments (Brown, Cavalier, & Tipton, 1986; Carr, 1989).

Speech Recognition, Phase One: Environmental Control

The first phase of the speech recognition research involved the development and field-testing of a prototype computer-based assistive device and accompanying training procedures that allowed a profoundly handicapped person to experience independent choice selection for the first time in her life. The participant in this research was a 42-year-old woman who lived in an institution most of her life, who possessed no self-help skills, who had almost completely unintelligible vocalizations, who was evaluated as profoundly mentally retarded with an IQ of 19, and who was confined to a bed or gurney-chair for the entire day. She learned through application of an innovative training procedure how to activate appliances and devices in her environment with vocalizations. The vocalizations did not even have to be words in the traditional sense; the computer system was programmed to respond to whatever utterances the woman could produce. This was the first time the use of such technology was ever attempted with a person with such severe handicaps (Brown, 1985).

Prior to Brown's research it had not been determined whether people with profound mental handicaps could understand the cause-and-effect relationships central to successful operation of such a system and learn to purposively use such technology, or if the benefits of such use would be substantial. Brown's results demonstrated the power and utility of such a system in increasing such individuals' functional control over their surroundings as well as improving their self-perception and the expectations of others around them.

Speech Recognition, Phase Two: Sound-to-Speech Translation

From the experiences with the woman who participated in the phase one research, the ARC identified additional features that would make the speech recognition computer system more useful and powerful. Therefore, the second phase of the speech recognition system resulted in the development of the Sound-to-Speech Translation (S-T-S) system prototype and software which adds communication output and new operational enhancements to the environmental control and VCR operation. The S-T-S translation system is designed as a direct selection assistive device and it has undergone multiple rounds of alpha testing in 1988. As a result of these tests, system refinements were incorporated in terms of speed of output, ease of training the voice templates, and increased accuracy of data recording and data reduction.

Research results from field testing indicate that the S-T-S translation system is technologically effective and well-designed. Behavioral data indicates that the subjects, who are severely mentally retarded and physically involved with only the rudiments of language, are learning to use the S-T-S system to begin to map out the cause-and-effect relationships in the environment are learning the practical referential meaning of single-syllable utterances for communication (Carr, 1989). From this foundation, the system can also be used to teach the fundamentals of combining two items to increase communication versatility.

Speech Recognition, Phase Three: Sound-to-Speech Recognition with Graphics

Although the S-T-S Translation system proved to be effective, areas for improvement of this first S-T-S translation prototype have been identified. Phase Three of the S-T-S device will maintain and build on the features of the Phase Two prototype with the exception of configuring phase three to accept generic VCR models. The device would be greatly enhanced and it would benefit more users, both sophisticated users and lower cognitively functioning users, if the following modifications were made.

The software is being redesigned so that the user can be presented with a menu selection of options which can be accessed with one or two vocalizations through a scanning process. The user or assistant can specify the speed of scanning. The user will have the option of directing the communication output to a printer in addition to, or in place of, the speech synthesizer. The system can be reconfigured to interface with a color scanner which will allow a user a defined array of photographic quality stimuli to be used in the menu array of selectable options mediated through a reconfigurable friendly dynamic visual display. The user can communicate in an "immediate" mode in which each single input is immediately output in its corresponding translation. The system will also communicate in a "delay" mode in which the system holds each successive input in a queue and then outputs the complete spoken translation upon command by the user. The software is being expanded to permit voice operation of a telephone.

Bioengineering Eyegaze Research

There are two main reasons for employing eyegaze-based and headtracking aids over more conventional body movement-based aids. First, a large portion of the population of individuals who are severely handicapped (those who are severely spastic and paralytic) do not have any other reliably controllable body movements. Second, even among those people with other reliable body movements, headtracking and especially the eye's response time (20-30 milliseconds) (Tello, 1988) are much more rapid and less fatiguing. Rapidity and ease of constructing a message are typically the determining factors between a person voluntarily using a communication aid and using it only when required to do so. The minimal expenditure of effort that eyegaze head movement entail and the high speed of use that they offer a person makes them tremendously promising for applications to people with severe handicaps. Recent developments in this technology now make such an access mode possible.

While eyegaze technology was initially developed for military purposes, since 1976 several researchers have conducted pilot projects on its potential to circumvent the problems associated with other technology approaches for people who are severely disabled (Demasco & Fould, 1982; Friedman, Kiliany, Dzmura, & Anderson, 1981; King, 1984; Rosen, Drinker & Dalrymple, 1976; Schneider, 1982; Sutter, 1983). This experimental research has been done only with severely motorically-impaired persons and the researchers agree that there is a need for a more concerted and diligent effort to further this body of knowledge (Angelo, Deterding, & Weisman, 1989; Smith, Christiaansen, Borden, Lindberg, Gunderson, & Vanderheiden, 1989).

Early in the development of eyegaze technology for assistive device application, Demasco and Foulds developed an ocular controlled device prototype and conducted research using a mechanical eye with a charged-coupled device video camera (Demasco, 1986; Demasco & Foulds, 1981; Demasco & Foulds, 1982). Their research gave positive indicators about the potential for further research and development in the area of eyegaze technology. Most products are specifically designed for either headpointing or eyegaze with the exception of the Eyetyper, which can do both.

Since 1987, findings and strategies developed from the voice recognition research have been extended into the arena of eyegaze/headpointing detection technology through research funded by the Office of Special Education Programs (Brown & Cavalier, 1989). This first phase of research has been devoted to answering fundamental questions about what features are needed in assistive technology that are controlled through a user's eyegaze or headpointing to open up the world of advanced technology to children and adults with severe mental retardation and severe physical impairments. The plan of the project was to design and develop the eyegaze and headtracking prototype hardware and software and to test the systems. The second and concurrent plan was to design and implement specific student training procedures to use with the students and teachers. Bioengineering staff have completed the eyegaze/headpointing prototype research and development and field testing in the Dallas and Arlington schools.

The ARC research staff adapted a Sentient Systems Eyetyper and through modification greatly enhanced this system and developed a new prototype/device so that:

- it has an enormously improved ability to accurately recognize where the user is looking;
- it collects data for each user as s/he interacts with the device for research and tracking purposes;
- the software has been modified so that a single device can be selectable not only for eyegaze but also for headpointing;
- the spoken output can be digitally recorded in any voice with the spoken output sounding like totally normal speech instead of synthetic speech;
- the speech output can be recorded on a computer and then transferred back to the device, thus multiple speech outputs can be saved and recalled at will;
- there is an increase in the display for the number of choices which can be displayed for the graphics;
- the prototype/device has the ability to control up to 256 electrical appliances in the environment;

- it is interfaced with a computer for system software upload and download which allows for individualized configuration of the system from one user to the other; and.
- it is totally portable with a 24 hour charge.

Research Findings

Preliminary evaluation of the research data and project materials from this eyegaze/headpointing research indicate several important findings.

- The subjects have learned to effectively use the technology. They display a variety of physical characteristics with varying degrees of spasticity, athetosis, hypertonia, and hypotonia. The proper seating of the subjects is critical to their ability to effectively utilize the technology with more of the subjects being effective headpointers than eyegazers.
- The subjects are benefiting educationally, socially and emotionally from the entire gestalt of the field testing, the technology, and the research implementation.
- The subjects' affective behavior primarily indicate involvement with the task of using the system, followed by a demonstration of pleasure at having control and decision-making ability.
- The hardware and software used for this research was developed from existing technology (*The Eyetyper*) and has been modified dramatically. The technology, even in a modified state, has been difficult and limited in its ability to address the needs of the subjects in this research project. Limitations in its ability to quickly and easily calibrate where the user is looking when first seated in front of the device is a major shortcoming. This is especially true for subjects with lower cognitive abilities.
- Significant training strategies can be developed for teachers, clinicians and parents on head positioning and control for use in future implementations with other users.

Summary

The ARC is excited about the ongoing research in assistive technology in the Bioengineering Program. The track of research on Freedom of Choice and Independence is demonstrating the incredible potential that technology can have for people with disabilities. And more importantly the abilities of the individuals with severe handicaps who use the technology and are now have the freedom to tell the world for the first time about those abilities and their individual choices.

References

Bricker, D., & Filler, J. (1985). The severely mentally retarded individual: Philosophical and implementation dilemmas. In D. Bricker & J. Filler (Eds.), *Severe mental retardation: From theory to practice* (pp. 2-11). Lancaster, PA:. Lancaster Press, Inc.

Brown, C.C. (1985). *Computerized voice recognition and environmental control with a person who is severely physically involved and profoundly mentally retarded.* Unpublished doctoral dissertation, North Texas State University.

Brown, C.C. (1989). *Freedom to speak and choose.* Presentation at the AAMR/AAC Augmentative Communication Colloquium, AAMR 113th Annual Meeting, Chicago, IL.

Brown, C.C., & Cavalier, A.R. (1989). *Freedom of choice and freedom of expression through an eyegaze-activated computerized device.* Unpublished document.

Brown, C.C., Cavalier, A.R., & Tipton, L. (1986). Increased independence through computerized voice recognition for persons who are severely physically involved and profoundly mentally retarded. In M. Donath, H. Friedman, & M. Carlson (Eds.), *Proceedings of the Ninth Annual Conference on Rehabilitation Technology,* (pp. 101-103). Minneapolis, MN: RESNA.

Carr, A.M. (1989). *Effects of a voice activated environmental control system on persons with severe/profound multiple disabilities.* Unpublished master's thesis, George Mason University, Fairfax, VA.

Demasco, P.W., & Foulds, R. (1981). Accuracy and linearity measurements of a CCD video camera for use in a line of gaze communication device. In J. Trimble, J. Doubler, & C. Heckathorne (Eds.), *Proceedings of the Fourth Annual Conference on Rehabilitation Engineering* (pp. 128-130). Washington, DC: RESNA.

Demasco, P.W., & Foulds, R.A. (1982). The design of an ocular controlled communication system for clinical implementation. In J.P. O'Leary & J.R. O'Reagan (Eds.), *Proceedings of the Fifth Annual Conference on Rehabilitation Engineering,* (p. 26).

Ellington, E.F. (1986). Computer-based toys for disabled children: A feasibility study. In M. Donath, H. Friedman, & M. Carlson (Eds.), *Proceedings of the Ninth Annual Conference on Rehabilitation Technology* (pp. 187-189). Minneapolis, MN: RESNA.

Esposito, L. (1985). Use of computer programs to train switch activation skills with young children with handicaps. In C. Brubaker (Ed.), *Proceedings of the Eighth Annual Conference on Rehabilitation Technology* (pp. 252-253). Memphis, TN: RESNA.

Fredericks, B. (1985). Parents/families of persons with severe mental retardation. In D. Bricker & J. Filler, (Eds.), *Severe mental retardation: From theory to practice* (pp. 142-160). Lancaster, PA: Lancaster Press, Inc.

Friedman, M.B., Kilinay, G., Dzmura, M., & Anderson, D. (1981). The eyetracker communication system. In *Proceedings of the Johns Hopkins First National Search for Applications of Personal Computing to Aid the Handicapped* (pp. 183-185). NY: IEEE.

King, M. (1984). Eyescan. *Communication Outlook, 5*(9).

Lazzaro, J.J. (1986, Dec). Talking instead of typing. *High Technology*, pp. 58-59.

Meeham, D.M., Mineo, B.A., & Lyon, S.R. (1985). Use of systematic prompting and prompt withdrawal to establish and maintain switch activation in a severely handicapped student. *Journal of Special Education Technology, 7*(1), 5-11.

Mineo, B.A., & Cavalier, A.R. (1989). *An examination of the appropriate medium for graphics representation for individuals with severe mental retardation.* Unpublished manuscript.

Parette, H.P., & Strother, P.O., & Hourcade, J. (1986). Microswitches and adaptive equipment for severely impaired students. *Teaching Exceptional Children, 19*(1), 15-18.

Romich, B. & Russell, N. (1985). An improved optical headpointer selection technique. In C. Brubaker (Ed.), *Proceedings of the Eighth Annual Conference on Rehabilitation Technology* (pp. 263-264). Memphis, TN: RESNA.

Rosen, M., Drinker, P., & Dalrymple, G. (1976). A display board for non-vocal communication encoded as eye movements. In *Proceedings of the Conference on Systems and Devices for the Disabled,* (pp. 70-71).

Schneider, M.R. (1982). Design of a new communication aid: SPA-SYN-COM. In J.P. O'Leary & J.R. O'Reagan (Eds.), *Proceedings of the Fifth Annual Conference on Rehabilitation Engineering,* (p. 25).

Smith, R.O., Christiaansen, R., Borden, B., Lindberg, D., Gunderson, J., & Vanderheiden, G. (1989). Effectiveness of a writing system using a computerized long-range optical pointer and 10-branch abbreviation expansion. *Journal of Rehabilitation Research and Development, 26*(1), 51-62.

Sutter, E.E. (1983). An oculo-encephalographic communication system. In B.R. Bowman (Ed.), *Proceedings of the Sixth Annual Conference on Rehabilitation Engineering* (pp. 171-173). San Diego: RESNA.

Tello, E. (1988, September). Between man and machine. *BYTE*, pp. 288-293.

Torner, R.S. (1986). A switch for education: Utilizing simplified microswitch technology. *Journal of Special Education Technology, 7*(4), 25-31.

Weintraub, L. (1987, December). An interview with Raymond Kurzweil. *Special and Individual Needs Technology (SAINT)*, pp. 1-4.

Wright, C., & Nomura, M. (1985). *From toys to computers: Access for the physically disabled child.* San Jose, CA: Authors.

In H. J. Murphy (Ed.), Proceedings of the Seventh Annual Conference on Technology and Persons with Disabilities (pp. 99-103). Northridge: California State University.

THE SOUND-TO-SPEECH TRANSLATION UTILIZING GRAPHIC SYMBOLS

Carrie Brown, PhD
Catherine Wyatt
The Arc

Maggie Sauer, M.S., CCC-SpL
University of North Carolina

Al Cavalier, PhD
University of Delaware

The Sound-to-Speech Translation Utilizing Graphic Symbols (STS/Graphics) system is a voice recognition computer-based system designed to allow individuals with mental retardation and/or severe physical disabilities to communicate with others and to control their environment. Operation of the STS/Graphics system is through speech/vocalizations, switch activation, or keyboard input for activation of electrical appliances and/or digitized speech output. Customized photographic quality symbols representing familiar items in the user's environment are displayed on the monitor for the user's choice making. Choices for communication or environmental control are made through direct selection, linear scanning, or row/column scanning. Assessment and educational applications are additional uses of the system. Research on the STS/Graphics system was conducted with students in a local public school system.

Background

There are more than two million children and adults in the United States whose physical and/or mental limitations are so severe that they are unable to communicate with other persons or interact with and learn from their environment in an effective manner. Over the past five years, voice recognition technology has advanced to the point where it now represents a viable, and even quite versatile, means of alternate access to education and habilitation for persons with severe handicaps. Speech is the most natural way to communicate and to act on the environment. Therefore, it is also a natural means by which to empower an individual. Computer technology, configured with voice-recognition access for communication and environmental control output, has the potential to effectively compensate for the limitations imposed by mental retardation and/or physical limitations. The Sound-to-Speech Translation Utilizing Graphic Symbols (STS/Graphics) system investigated the use of speech technology for communication and environmental control.

The STS/Graphics system is a voice recognition computer-based system designed to allow individuals with mental retardation and/or severe physical handicaps to communicate with others and to control their environment. This includes individuals with quadriplegia, spinal cord injury, cerebral palsy, aging problems, arthritis, and assorted neurological disorders. Operation of the system is through speech/vocalizations, switch activation, or keyboard input for operation of electrical appliances and/or digitized speech output. Customized photographic-quality symbols representing available choices are displayed on the monitor for the user. The choices for communication or environmental control are selectable through direct selection, linear scanning, or row/column scanning. Assessment and educational applications are other uses of the system.

The main features of the system are as follows:

Voice Input. The system incorporates the Votan 2000 voice recognition circuit board and software routines. Each user can store voice templates for sets of up to 64 messages, consisting of sounds or word phrases of up to 8 seconds each in duration. Multiple users can simultaneously store templates for their messages and access them. For individuals for whom consistent sound production is not possible, the system can be adapted to accept a simple vocalization as a means of selecting desired items via a visual scanning routine.

Sound-to-Speech Translation and Expansion. The system incorporates an algorithm that translates any designated sound input, whether intelligible or not, into a specified speech output. For example, if a user's vocalization for water approximates "wuh" and s/he only speaks in single-syllable vocalizations, the system can immediately output "Could I have a drink of water, please." The output is digitized speech which can be up to 8 seconds in duration, thus permitting single words or complete sentences to be output.

Environmental Control. The system incorporates X-10 technology. Any voice (or other) input can be linked to the activation and deactivation of any electrical device. A single voice input can be linked to any combination of spoken outputs and device activations.

Universal Controller. The system uses a One-for-All universal controller which can control any audio/visual device that operates with the use of an infra remote controller.

Graphics System Interface. The system generates photographic-quality images of items, appliances, or people in the user's environment. In addition, the size of the image as well as the number of images appearing on the display, can be customized to the user. This is achieved by digitally scanning the images into the computer. For example, a photograph of a tape player might be configured to produce the message "I like this music" as well as activating the tape recorder to play music. Displays can be created of varying sizes ranging from a 2 x 2 to a 5 x 5 matrix. The graphics display can "flip" to a second page of choices as well.

The STS/Graphics software provides two methods of accessing the system: direct selection, and scanning. In the direct select mode, a user produces different sounds, each of which is matched to a picture that may be included in a single or multi-picture display. A set of sounds can be "trained" in the system and stored for each potential user. A picture can be associated with different utterances for each user. Thus, the system is customized for each person who uses it. Scanning can be operated two ways: linear scanning or row-column scanning. In either scan mode, the system requires only a single vocalization, and it is not necessary for this sound to be very consistently pronounced from one instance to the next. This vocalization is used to halt the scan routine. Once the scan is halted, the functions that are associated with the cell on which the scan stopped are executed: that is, speech output, environmental control, or both.

Photo libraries are created for the users. A library consists of disks containing scanned images of items which have personal meaning to the user. The library is arranged by category and may serve as a good organizational model for the users. In this way, images may be used for a variety of subjects in their displays.

<u>Software Design</u>. The software package is made up of three major components: Speech Manager, Setup, and Sound-to-Speech. **Speech Manager** provides the audio recording and vocabulary training functions for the speech recognition board. Speech Manager processes and stores the utterances or words that are used as a trigger or switch by the software. For use in the Direct Selection method of access, the vocabulary trainings are matched to an individual picture. In the Scanning access method, only one sound training is in the system. The **Setup** section of the software is used to combine picture files, audio output messages, and environmental control commands into specific cells in a matrix. When all of the required cells in a matrix are filled, then the information is stored as a "page" file. **Sound-to-Speech** executes any of the pages created by Setup and generates activity reports regarding the operation of the software.

<u>Reports</u>. The system generates activity reports regarding the operation of the software. As each cell is activated, the actions corresponding to that cell, such as audio playback and environmental control, are stored in a data file from which reports are derived containing analyses of these activities.

Research Questions

The following research questions were addressed in this study: Can the subject learn the cause/effect relationship between vocalization and function? Can the subject learn that different vocalizations are associated with different functions? Will the subject exhibit increased communication with others? Is there a degree of mental retardation that excludes a user from effective operation of this device? Is the graphics component a viable option for this type of communication/environmental control device? What are the strengths and limitations of the system?

Method

The hardware used in this study were a PC/AT microcomputer, a Votan VPC 2000 Speech Card and Speech Recognizer, a VGA Color Monitor, a Color Scanner, and a Mouse. The software was customized according to the earlier description.

Subjects. Twenty-four students from the local public schools were evaluated. The characteristics of the six selected subjects were as follows: four females, two males, four ambulatory, two non-ambulatory, ages 7-21, with diagnosis of cerebral palsy, encephalitis, and mild to profound mental retardation. The subjects understood rudimentary scanning, knew cause and effect, and had adequate vision and hearing. Research was completed on three of the subjects. Two were direct selectors and ambulatory; one was a scanner and non-ambulatory.

Procedure. The STS/Graphics system was used in the school environment of each subject. Prior to the experimental trials, personal items presented on the graphics display were identified using forced choice selection with each subject for communication, environmental control, or a combination of both. Identification of each subject as a direct selector or as a scanner was determined. Operation of the system was integrated with specific training strategies used with the subjects to teach them proper delivery of personal vocalizations for system training and operation as well as for choice making from items on the display. Each subject progressed through specific training steps from selecting a single item on the display to selecting multiple items on the display which were topically similar when presented (i.e., beauty aids). A trial consisted of the researcher prompting the subject to activate a specific device or communication phrase. A trial was deemed successful if the subject emitted the vocalization that corresponded with the device or communication that was prompted, and if the computer system recognized it as such. During single item trials, there was only one graphics icon displayed on the monitor. Over time, additional graphics icons and their corresponding choices were added to the display. The subject would advance from the single item to multiple items based on successfully completing nine out of twelve trials on each level of item displayed.

<u>Research Design</u>. A multiple-baseline-across-target-vocalizations experimental design was implemented with all subjects, whereby designated vocalizations resulted in output of a communicated phrase or operation of an electrical device in the environment.

Results

The three subjects, each of whom had different degrees of mental retardation, learned to communicate and to make choices by using their own personal vocalizations to operate the STS/Graphics system. There was variability among the subjects in their level of proficiency in operating the system. All learned the cause/effect relationship between vocalization and action, and also learned that different vocalizations were associated with different functions. The subjects demonstrated increased frequency of vocalization and speech usage outside the research environment. The graphics component of the system was versatile and appropriate in this application.

Discussion

The STS/Graphics system is a viable method for communication and environmental control by persons having mental retardation and/or other physical disabilities. Limitations of the STS/Graphics system were the microphone placement demands, additional training of the system due to subjects improved vocalization, noisy environments, and a lack of portability of the system. Additional applications of this system for evaluation, training, and education should be explored. The system was effective in encouraging persons with mental retardation to use their natural speech to communicate and to control their environment. Further research on voice recognition for persons with mental retardation should be conducted.

Acknowledgements

Funding for research was provided by the U.S. Dept. of Education, Grant No. H180P90015, The Arc Research Fund, and Martin Marietta Corporation.

A documentary videotape of this research, Voices of the Future, is available from The Arc. A detailed operations and training manual accompanies the Sound-to-Speech/Graphics system.

(1992). Augmentative and Alternative Communication, 8(2).

Speech Recognition and Graphics Research for Persons Having Mental Retardation

Carrie Brown, PhD, Director, Bioengineering Program, Association for Retarded Citizens of the United States, 500 E. Border, Suite 300, Arlington, TX 76010

Maggie Sauer, CCC-SPL, Assistant Clinical Professor, Clinical Center for Development and Learning, CB7255 BSRC, University of North Carolina, Chapel Hill, NC 27599

Al Cavalier, PhD, Associate Professor, Department of Educational Studies, Willard Hall, 213-A, University of Delaware, Newark, DE 19711

Catherine Wyatt, Technology Coordinator, Bioengineering Program, Association for Retarded Citizens of the United States, 500 E. Border, Suite 300, Arlington, TX 76010

The Sound-to-Speech Translation Utilizing Graphic Symbols system is a voice recognition computer-based system designed to allow individuals with mental retardation and/or severe handicaps to communicate with others and to control their environment. Operation is through speech/vocalizations, switch activation and keyboard input for operation of electrical appliances and/or digitized speech output. Customized photographic-quality symbolic choices, displayed on the monitor, are available. Choices for communication or environmental control are selectable through direct selection, linear scanning, or row/column scanning. Assessment and educational applications are additional uses of the system.

System Features

Speech Output. The system incorporates a voice recognition circuit board and unique software routines which allows digitized audio messages to be recorded, stored and played back for multiple users. Output is digitized speech which can be up to 8 seconds in duration allowing single words or complete sentences output.

Sound-to-Speech Translation and Expansion. Vocalization specific to each user are stored in templates. Multiple users can simultaneously store templates which are matched to speech output messages. For individuals for whom consistent sound production is not possible, the system can be adapted to accept simple vocalizations as a means of selecting desired items from a visual scanning routine. The system incorporates an algorithm that translates any designated sound input, whether intelligible or not, into a specified speech output. For example, if a user's vocalization for water approximates "wuh" and s/he only speaks in single-syllable vocalizations, the system immediately outputs "Could I have a drink of water, please."

Environmental Control. The system incorporates X-10 technology by voice (or other) input links to the activation and deactivation of any electrical device. Single voice input links any combination of spoken outputs and device activations.

Universal Controller. The system accesses a universal controller for control of any audio/visual device that operates through an infrared remote controller.

Graphics System Interface. The system displays photographic-quality images of items, appliances or people in the user's environment. The size of the image, as well as the number of images appearing on the display, are customized to the user. This is achieved by using an image capturing device such as a color scanner or camera to digitize the images into the computer. Displays can be created in varying sizes ranging from 2 x 2 to a 5 x 5 matrix. The graphics display also "flips" to a second page of choices.

The software provides two methods of access to the system: Scanning and Direct Selection. In the Scanning mode, user activations are triggered by a single utterance or standard switch. In Direct Select mode, multiple unique sounds are matched to a picture that is included in a single or multi-picture display. A set of sounds is "trained" in the system and stored and customized for each user. An image can be associated with different utterances for each user.

Scanning is through linear scanning or row-column scanning. In either mode, the system only requires a single vocalization, and the utterance produced does not need to be very similar on a consistent basis. If speech is not possible, a single switch can replace voice activation. Photo libraries are created for the users containing scanned images of personalized items.

Reports

The system generates activity reports regarding the software operation. As each cell is activated, actions corresponding to that cell, such as audio playback and environmental control, are stored in a data file for report generation. These reports detail and analyze specific activities such as environmental control and audio playback.

Training Manual

A detailed Operations and Training Manual accompany the Sound-to-Speech/Graphics system, especially tailored for use with people having mental retardation.

Field Testing

Research and data was collected on subjects having mental retardation in a local school district. Research findings will be presented and discussed.

Funded By: U.S. Dept. of Education - Grant #H180P90015
ARC Research Fund
Martin Marietta

Sound-to-Speech Translation Utilizing Graphic Symbols

Carrie Brown, PhD, The Arc, Arlington, TX
Maggie Sauer, M.S.,CCC-SpL, University of North Carolina, Chapel Hill, NC
Al Cavalier, PhD, University of Delaware, Newark, DE
Catherine Wyatt, The Arc, Arlington, TX

Abstract

The Sound-to-Speech Translation Utilizing Graphic Symbols (STS/Graphics) system is a voice recognition computer-based system designed to allow individuals with mental retardation and/or severe physical disabilities to communicate with others and to control their environment. Operation of the STS/Graphics system is through speech/vocalizations, switch activation, or keyboard input for activation of electrical appliances and/or digitized speech output. Customized photographic quality symbols representing familiar items in the user's environment are displayed on the monitor for the user's choice making. Choices for communication or environmental control are made through direct selection, linear scanning, or row/column scanning. Assessment and educational applications are additional uses of the system. Research on the STS/Graphics system was conducted with students in a local public school system.

Background

There are more than two million children and adults in the United States whose physical and/or mental limitations are so severe that they are unable to communicate with other persons or interact with and learn from their environment in an effective manner (Bricker & Filler, 1985). Over the past five years, voice recognition technology has advanced to the point where it now represents a viable, and even quite versatile, means of alternate access to education and habilitation for persons with severe handicaps. Speech is the most natural way to communicate and to act on the environment. Therefore, it is also a natural means by which to empower an individual. Computer technology, configured with voice-recognition access for communication and environmental control output, has the potential to effectively compensate for the limitations imposed by mental retardation and/or physical limitations. The Sound-to-Speech Translation Utilizing Graphic Symbols (STS/Graphics) system investigated the use of speech technology for communication and environmental control.

The STS/Graphics system is a voice recognition computer-based system designed to allow individuals with mental retardation and/or severe physical handicaps to communicate with others and to control their environment. This includes individuals with quadriplegia, spinal cord injury, cerebral palsy, aging problems, arthritis, and assorted neurological disorders. Operation of the system is through speech/vocalizations, switch activation, or keyboard input for operation of electrical appliances and/or digitized speech output. Customized photographic-quality symbols representing available choices are displayed on the monitor for the user. The choices for communication or environmental control are selectable through direct selection, linear scanning, or row/column scanning. Assessment and educational applications are other uses of the system.

The main features of the system are as follows:

Voice Input. The system incorporates the Votan 2000 voice recognition circuit board and software routines. Each user can store voice templates for sets of up to 64 messages, consisting of sounds or word phrases of up to 8 seconds each in duration. Multiple users can simultaneously store templates for their messages and access them. For individuals for whom consistent sound production is not possible, the system can be adapted to accept a simple vocalization as a means of selecting desired items via a visual scanning routine.

Sound-to-Speech Translation and Expansion. The system incorporates an algorithm that translates any designated sound input, whether intelligible or not, into a specified speech output. For example, if a user's vocalization for water approximates "wuh" and s/he only speaks in single-syllable vocalizations, the system can immediately output "Could I have a drink of water, please." The output is digitized speech which can be up to 8 seconds in duration, thus permitting single words or complete sentences to be output.

Environmental Control. The system incorporates X-10 technology. Any voice (or other) input can be linked to the activation and deactivation of any electrical device. A single voice input can be linked to any combination of spoken outputs and device activations.

Universal Controller. The system uses a One-for-All universal controller which can control any audio/visual device that operates with the use of an infra remote controller.

Graphics System Interface. The system generates photographic-quality images of items, appliances, or people in the user's environment.

In addition, the size of the image as well as the number of images appearing on the display, can be customized to the user. This is achieved by digitally scanning the images into the computer. For example, a photograph of a tape player might be configured to produce the message "I like this music" as well as activating the tape recorder to play music. Displays can be created of varying sizes ranging from a 2 x 2 to a 5 x 5 matrix. The graphics display can "flip" to a second page of choices as well.

The STS/Graphics software provides two methods of accessing the system: direct selection, and scanning. In the direct select mode, a user produces different sounds, each of which is matched to a picture that may be included in a single or multi-picture display. A set of sounds can be "trained" in the system and stored for each potential user. A picture can be associated with different utterances for each user. Thus, the system is customized for each person who uses it. Scanning can be operated two ways: linear scanning or row-column scanning. In either scan mode, the system requires only a single vocalization, and it is not necessary for this sound to be very consistently pronounced from one instance to the next. This vocalization is used to halt the scan routine. Once the scan is halted, the functions that are associated with the cell on which the scan stopped are executed: that is, speech output, environmental control, or both.

Photo libraries are created for the users. A library consists of disks containing scanned images of items which have personal meaning to the user. The library is arranged by category and may serve as a good organizational model for the users. In this way, images may be used for a variety of subjects in their displays.

Software Design. The software package is made up of three major components: Speech Manager, Setup, and Sound-to-Speech. Speech Manager provides the audio recording and vocabulary training functions for the speech recognition board. Speech Manager processes and stores the utterances or words that are used as a trigger or switch by the software. For use in the Direct Selection method of access, the vocabulary trainings are matched to an individual picture. In the Scanning access method, only one sound training is in the system. The Setup section of the software is used to combine picture files, audio output messages, and environmental control commands into specific cells in a matrix. When all of the required cells in a matrix are filled, then the information is stored as a "page" file. Sound-to-Speech executes any of the pages created by Setup and generates activity reports regarding the operation of the software.

Reports. The system generates activity reports regarding the operation of the software. As each cell is activated, the actions corresponding to that cell, such as audio playback and environmental control, are stored in a data file from which reports are derived containing analyses of these activities.

Research Questions

The following research questions were addressed in this study: Can the subject learn the cause/effect relationship between vocalization and function? Can the subject learn that different vocalizations are associated with different functions? Will the subject exhibit increased communication with others? Is there a degree of mental retardation that excludes a user from effective operation of this device? Is the graphics component a viable option for this type of communication/environmental control device? What are the strengths and limitations of the system?

Method

The hardware used in this study were a PC/AT microcomputer, a Votan VPC 2000 Speech Card and Speech Recognizer, a VGA Color Monitor, a Color Scanner, and a Mouse. The software was customized according to the earlier description.

Subjects. Twenty-four students from the local public schools were evaluated. The characteristics of the six selected subjects were as follows: four females, two males, four ambulatory, two non-ambulatory, ages 7-21, with diagnosis of cerebral palsy, encephalitis, and mild to profound mental retardation. The subjects understood rudimentary scanning, knew cause and effect, and had adequate vision and hearing. Research was completed on three of the subjects. Two were direct selectors and ambulatorypry; one was a scanner and non-ambulatory.

Procedure. The STS/Graphics system was used in the school environment of each subject. Prior to the experimental trials, personal items presented on the graphics display were identified using forced choice selection with each subject for communication, environmental control, or a combination of both. Identification of each subject as a direct selector or as a scanner was determined. Operation of the system was integrated with specific training strategies used with the subjects to teach them proper delivery of personal vocalizations for system training and operation as well as for choice making from items on the display. Each subject progressed through specific training steps from selecting a single item on the display to selecting multiple items on the display which were topically similar when presented (i.e., beauty aids). A trial consisted of the researcher prompting the subject to activate a specific device or communication phrase. A trial was deemed successful if the subject emitted the vocalization that corresponded with the device or communication that was prompted, and if the computer system recognized it as such. During single item trials, there was only one graphics icon displayed on the monitor. Over time, additional graphics icons and their corresponding choices were added to the display. The subject would advance from the single item to multiple items based on successfully completing nine out of twelve trials on each level of item displayed.

Research Design. A multiple-baseline-across-target-vocalizations experimental design was implemented with all subjects, whereby designated vocalizations resulted in output of a communicated phrase or operation of an electrical device in the environment.

Results

The three subjects, each of whom had different degrees of mental retardation, learned to communicate and to make choices by using their own personal vocalizations to operate the STS/Graphics system. There was variability among the subjects in their level of proficiency in operating the system. All learned the cause/effect relationship between vocalization and action, and also learned that different vocalizations were associated with different functions. The subjects demonstrated increased frequency of vocalization and speech usage outside the research environment. The graphics component of the system was versatile and appropriate in this application.

Discussion

The STS/Graphics system is a viable method for communication and environmental control by persons having mental retardation and/or other physical disabilities. Limitations of the STS/Graphics system were the microphone placement demands, additional training of the system due to subjects improved vocalization, noisy environments, and a lack of portability of the system. Additional applications of this system for evaluation, training, and education should be explored. The system was effective in encouraging persons with mental retardation to use their natural speech to communicate and to control their environment. Further research on voice recognition for persons with mental retardation should be conducted.

Bricker, D., & Filler, J. (1985). The severely mentally retarded individual: Philosophical and implementation dilemmas. In D. Bricker & J. Filler (Eds.), Severe mental retardation: From theory to practice (pp. 2-11). Lancaster, PA: Lancaster Press, Inc.

Acknowledgements

Funding for research was provided by the U.S. Dept. of Education, Grant No. H180P90015, The Arc Research Fund, and Martin Marietta Corporation.

A documentary videotape of this research, *Voices of the Future*, is available from The Arc. A detailed operations and training manual accompanies the Sound-to-Speech/Graphics system.

Carrie Brown, PhD, Director
Bioengineering Program
The Arc
500 East Border, Suite 300
Arlington, TX 76010

Reprinted from

The Johns Hopkins APL Technical Digest

CARRIE BROWN, AL CAVALIER, MAGGIE SAUER, AND CATHERINE WYATT

THE SOUND-TO-SPEECH TRANSLATION SYSTEM USING PHOTOGRAPHIC-QUALITY GRAPHIC SYMBOLS

The Sound-to-Speech Translation System Using Photographic-Quality Graphic Symbols (STS/Graphics System) is a voice-recognition computer-based system designed to allow people with mental retardation and/or severe physical disabilities to communicate with others and to control their environment. This includes people with quadriplegia, spinal cord injury, cerebral palsy, aging problems, arthritis, and various neurological disorders. Operation of the STS/Graphics System is through vocalization, switch closure, or keyboard input for activation of electrical appliances and/or digitized speech output. Choices available to the user for system operation are presented as customized photographic-quality symbols representing familiar items/persons in the user's environment that are displayed on a monitor. The user makes choices through direct selection, linear scanning, or row/column scanning. The system represents a powerful and versatile means by which children and adults with severe disabilities can increase their independence and productivity. Assessment and educational applications are additional uses of the system.

INTRODUCTION

The American Association on Mental Retardation states that

> Mental retardation refers to substantial limitations in present functioning. It is characterized by significantly subaverage intellectual functioning, existing concurrently with related limitations in two or more of the following applicable adaptive skill areas: communication, self-care, home living, social skills, community use, self-direction, health and safety, functional academics, leisure and work. Mental retardation manifests before age 18.[1]

A person with mental retardation may need one of four possible levels of support: (1) intermittent, does not require constant support; (2) limited, requires certain supports consistently over time; (3) extensive, needs daily support in some aspects of daily living; and (4) pervasive, requires constant, high-intensity support for all aspects of life.[2] Although information is limited on the number of people with mental retardation who do not communicate orally, a recent study in the state of Washington estimated that about 4% to 12% of school-aged children with mental retardation needing intermittent to limited support, and 92% to 100% of people with mental retardation needing extensive to pervasive support, were nonvocal communicators.[3] In total, more than two million children and adults in the United States have mental and/or physical limitations so severe that they are unable to communicate effectively with other persons without augmentative or alternative communication assistance.[4] As defined by the American Speech-Language-Hearing Association,

> Augmentative and alternative communication (AAC) is an area of intervention in which attempts are made to compensate and facilitate, temporarily or permanently, for the impairment and disability patterns of individuals with severe expressive, and/or language comprehension disorders.[5]

Many users of AAC devices have multiple handicaps, that is, they might have any combination of severe physical impairments, sensory impairments, and mental retardation.[6] Augmentative and alternative communication systems for such people can be simple or sophisticated—for example, a cardboard sheet with pictures that depict communication needs or a computer that produces speech. Major contemporary issues in AAC research involve defining the parameters that facilitate the optimal use of these communication systems. Additionally, in the design of an AAC system, consideration is given to identifying the optimal access methods, type of speech output, and additional functions and features beyond speech, such as environmental control and data recording and analysis.

Recent studies suggest that the use of speech-output communication systems can have a significant effect on the communication skills of nonspeaking persons with cognitive disabilities.[7] Defining the parameters for optimal use by people with mental retardation needing extensive to pervasive support can be especially difficult, however, because of their delayed language comprehension and expression, the variation in their ability to process information, and their limited ability to understand the meaning of symbolic representations.[8] One of the most critical and challenging issues in using AAC systems with people with mental retardation is the development of effective teaching strategies to ensure that the person can successfully and flexibly use the system.[9]

During the past five years, voice-recognition technology has advanced to the point where it now represents a viable and versatile means of alternative access to education and habilitation for persons with severe disabilities.[10] Speech is the most natural way to communicate and to participate in the environment; therefore, it is also a natural means by which to empower a person. A growing body of research has been conducted on the use of voice-recognition technology with people who have normal cognitive abilities but who also have unintelligible or no speech.[11,12] On the basis of this research, it is reasonable to believe that computer technology configured with voice-recognition access for communication and environmental control has the potential to compensate effectively for the limitations imposed by mental retardation and physical disabilities. Relatively little research, however, has been attempted thus far with voice-recognition technology that addresses the needs of the population subject to those limitations.[13,14]

The research described herein involves the design and development of a voice-operated AAC computer system that was used by children and young adults with mental retardation to communicate and to control their environment through no greater effort than the vocalization of simple sounds.

THE SOUND-TO-SPEECH/GRAPHICS SYSTEM

The Sound-to-Speech Translation System Using Photographic-Quality Graphic Symbols (STS/Graphics System) is designed to explore the potential of voice-recognition technology for persons with mental retardation. It is an extension of previous research and development conducted by Brown and Cavalier and their colleagues at The Arc. (The Arc was formerly the Association for Retarded Citizens of the United States.) In earlier research, a voice-activated computer system was developed and evaluated with a woman diagnosed with mental retardation and severe multiple disabilities, who needed pervasive support. The woman learned to voice activate the system to control electrical devices in her immediate environment.[13,15] This study was followed by research on a multi-user system with environmental control and speech output capabilities.[16] Neither of these studies employed a picture-based display to assist the user in making choices.

The next generation of this research track, the STS/Graphics System, incorporates a graphics display. Key features of the STS/Graphics System are as follows: (1) photographic-quality computer-generated images of items/persons in the user's natural environment that are presented on a display to represent choices among spoken communications and environmental control activations; (2) vocalizations, whether intelligible or not, that are used to select choices from the image display; (3) physical switch activations and traditional keyboard inputs that are alternative access modes; (4) voice and keyboard access modes to make choices via direct selection; (5) voice and switch input to make choices via linear scanning and row/column scanning; (6) speech output that is in the form of digitized, easily understood speech and that is age and sex appropriate; and (7) data on system use that are automatically recorded, thereby facilitating the development of appropriate training strategies.

Hardware

The following list of hardware components represents the minimum configuration required for the operation of the STS/Graphics System. Enhanced performance can be obtained with a 386 or 486 microprocessor and greater hard disk capacity. Additional random access memory will not enhance performance.

1. IBM-PC advanced technology (AT) or compatible with a 286 central processing unit.
2. 30 MB of hard disk space.
3. Super vector graphics adapter monitor.
4. Paradise graphics board.
5. Votan 2000 voice-recognition board.
6. Image capturing device with color capability.
7. X-10 Powerhouse controller and various wall modules.
8. 640 KB of random access memory.
9. Floppy disk drive.
10. MS-DOS 3.30 or greater.

A schematic diagram of these components with enhancements is shown in Figure 1.

Voice Input

The voice-recognition system incorporates the Votan 2000 voice-recognition circuit board (Votan, Inc., Fremont, Calif.) and software routines unique to the system developed at The Arc. When operating the STS/Graphics System, the user's vocalizations function as switches, and the user "trains" the computer by speaking the target vocalizations into the microphone. This sound is recorded digitally on the hard disk. The training is repeated three times, and each vocalization is recorded separately. After the three trainings are stored in a user-specific voice template, the STS/Graphics System software analyzes the three recordings and averages them. For scanning access to the system (see the discussion in the Access Modes section), the user-specific template is composed of trainings for one unique utterance. For direct-selection access to the system (see the discussion in the Access Modes section), the user-specific template is composed of multiple unique vocalization trainings. After all of the trainings are stored, the system verifies that the user's target vocalization matches one of the sounds stored in the voice template. If the match is not similar enough to the original trainings, then the user must retrain the voice template for that particular utterance.

When the user is actually operating the system and produces a vocalization, the system checks to see if the vocalization matches any of the trained vocalizations stored in the voice template. If a match occurs, the system acknowledges the vocalization by activating the appropriate speech output message or environmental control function; if the vocalization does not match any of the trainings closely enough, the system will not respond. If the system does not respond, two optional corrective actions

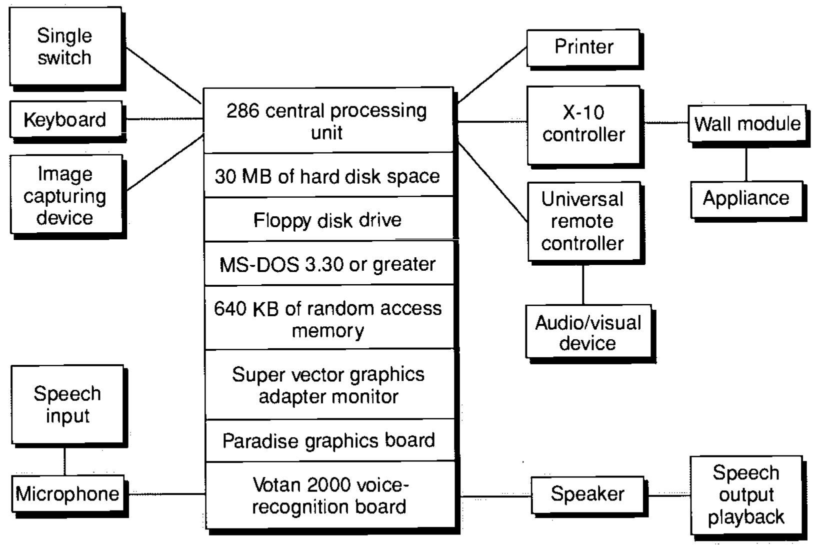


Figure 1. Hardware configuration of the Sound-to-Speech/Graphics System.

can be taken. First, the accuracy used by the system to match a vocalization to the stored voice trainings can be made less sensitive so that the system is less demanding in its matching requirements. Second, the user may choose to repeat the training process to refine a vocalization stored in the template for better recognition accuracy.

Sound-to-Speech Translation and Expansion

For communication purposes, each user can access speech output messages stored in digitized files on the hard disk. Each message can consist of speech of up to eight seconds in duration, thus permitting single words or complete sentences to be output. Someone whose voice is age- and sex-appropriate for the user records these messages into the system. The Votan board manages this process.

The system uses an algorithm to compare a user's vocalization with those previously stored in a voice template. If a match is found, the input is identified and linked to one of the speech output messages. In a sense, the user's often brief and unintelligible vocalization translates into intelligible speech and expands into a phrase or complete sentence. For example, if a user's vocalization for "water" approximates "wuh," the system can immediately output "Could I have a drink of water, please." Message files for many different users can be stored and accessed via one computer system. The number of available files depends only on the size of the system's hard disk.

Environmental Control

The system incorporates X-10 Powerhouse environmental control technology (X-10, Inc., Northvale, N.J.). The computer sends digital encoded signals over existing electrical wiring in the room or building in search of a lamp or wall module with the same identification code. The modules plug into wall outlets. The target module responds to the control signals transmitted from the computer and activates or deactivates the appliance plugged into it. Any voice (or other) input can be linked to the activation and deactivation of designated electrical devices. A single voice input can also be linked to any combination of speech outputs and device activations; for example, the vocalization "pa" can translate into "I'm going to watch TV now," and the television set turns on.

The system also uses a One-for-All universal controller (Universal Electronics, Inc., Signal Hill, Calif.) to control any audio/visual device that operates through the use of an infrared remote controller. The universal remote controller operates via a serial input/output port located inside its battery compartment. Using a simple communications protocol, the computer system, through an RS-232 port, electronically presses the buttons of the remote control unit to operate a device.

Graphics Interface

On the video monitor, the system displays photographic-quality images of objects, appliances, and people in the user's environment. The sizes of the images, as well as the number of images appearing on the display, are customized to the user. This customization is achieved by digitally scanning photographs of items into the computer using an image-capturing device (a scanner or video camera) and then sizing the image using image-manipulation software such as PC Paintbrush. Extensive image libraries are created for the users. An image library consists of disks containing scanned images of items that have personal meaning to the user. The library is arranged by category and serves as an organizational model for the teacher or trainer. In this way, the same images are used by a variety of subjects in their customized displays.

Displays range from a single image to a 5×5 matrix of images. On a thirteen-inch monitor, the size of each of the four images in a 2×2 matrix is about 3.75×5 in.; the size of each of the twenty-five images in a 5×5 matrix is about 1.5×2 in. The graphics display can also "flip" to a second page of choices for each user. The two pages of images permit the configuration of a hierarchical system of choices for users who understand the concept of categorical clustering. In such a system, the images displayed on page one represent superordinate categories, for example, an image for the food category, an image for the clothing category, and an image for the sports category. The selection of a page-one image displays a second page of images all related to the particular category selected on page one. It is not mandatory, however, that images on page one represent superordinate categories for the second page. On the second page, one cell is always designated as a "return" to page one.

The graphics images in this system act as cognitive prosthetic aids that remind the user of the speech outputs and device activations available. Whereas a user's voice input (or switch or keyboard input) initiates the sequence of electronic events that results in some functional output, the graphics images provide a symbolic representation of output choices that are "live," or available, at any one time.

Access Modes

The STS/Graphics System software provides two methods of accessing the system: direct selection and scanning. In the direct-selection mode, a user vocalizes a different sound for each image presented on the display. The scanning mode is operated in two ways: linear scanning or row/column scanning. In either scan mode, the system requires only a single vocalization to access all choices, and the vocalization does not have to be pronounced consistently from one occurrence to the next. The vocalization serves to stop the scanning sequence. Once the scan halts, the functions associated with the image on which the scan stopped are executed (i.e., speech output, environmental control, or both). Alternatively, the scanning can be stopped by using a single switch.

Software Design

The customized system software has three major components: speech manager, setup, and sound-to-speech, as shown in Figure 2.

The speech manager has two discrete functions: the record function and the recognition function. The record function, shown in Figure 3, creates the speech output message files and groups these audio messages into categories appropriate for the user or setting. New categories and messages can quickly and easily be added or deleted. Each audio message has a corresponding text description in its storage file. The recognition function, shown in Figure 4, creates and manages the voice template that contains the user's recordings of single or multiple vocalizations. In the direct-selection mode, a different vocalization is linked to each image on the display. In the scanning mode, only one vocalization is stored in the system, and it serves to halt the scan sequence on the desired image, resulting in a selection.

The setup component of the software, shown in Figure 5, is used to combine image files, audio output messages, environmental control, and infrared commands and to assign them to specific cells in a matrix (from 1×1 to 5×5). When all of the desired cells in a matrix are filled with images (it is not mandatory to put an image in each

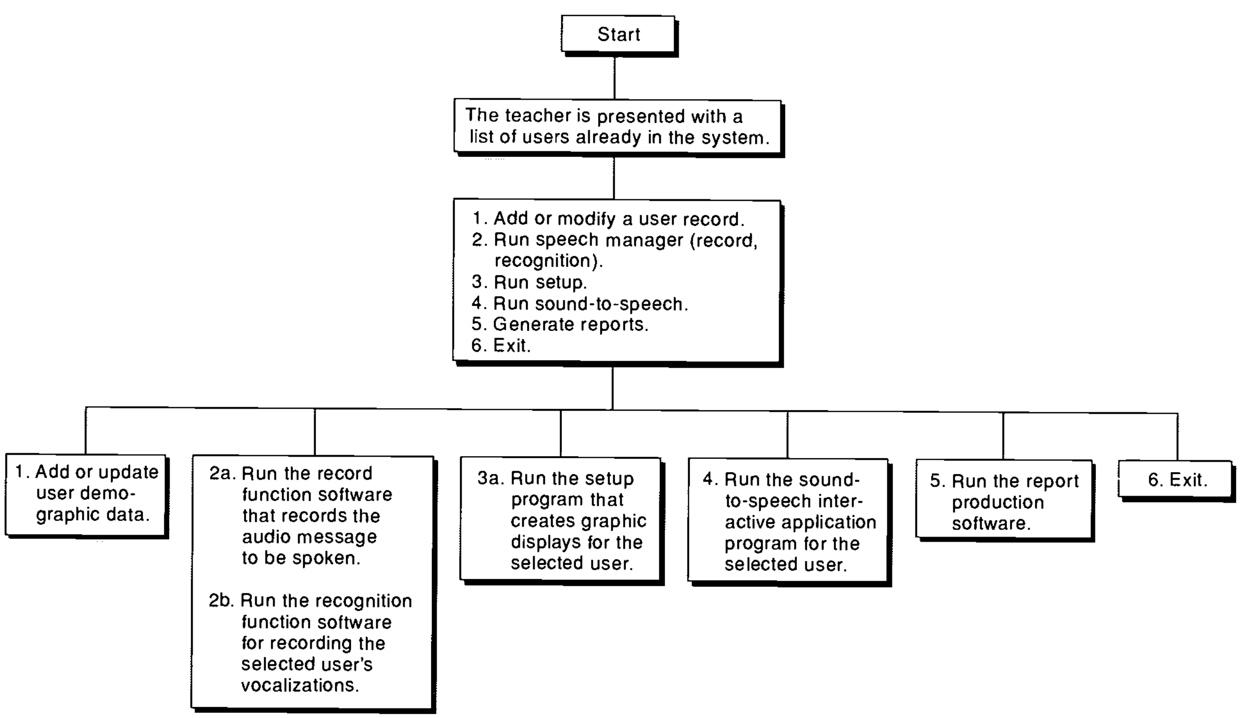


Figure 2. Main menu of the Sound-to-Speech/Graphics System software.

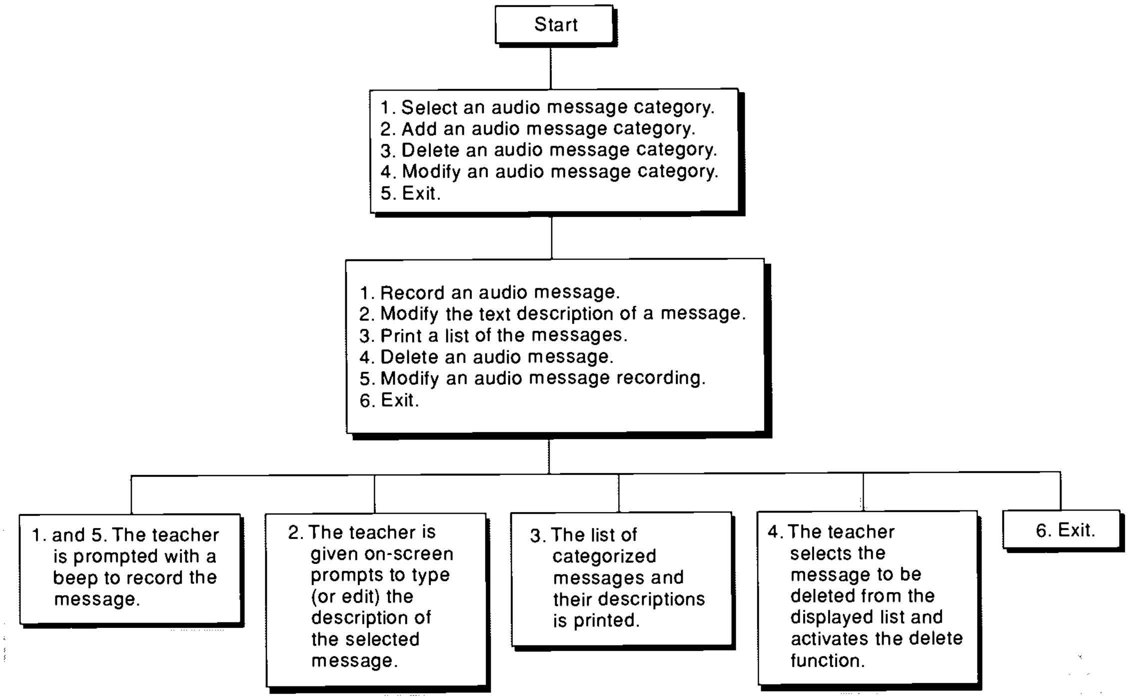


Figure 3. Record function of the speech manager component of the Sound-to-Speech/Graphics System software.

cell), then the information is stored as a "page" file. Each page contains different graphics choices, and the user can flip between two pages, as explained in the Graphics Interface section.

Once the system has been configured for use, the sound-to-speech component presents the pages to the user, as shown in Figure 6. It is the interactive portion of the system and the only one with which a user interacts for communication and environmental control. Data collection on user activity occurs whenever the sound-to-speech component is active. The collected data can be output as a report.

Reports

The STS/Graphics System generates activity reports regarding its operation. As each user's choice is activated, a record of the interaction is written to a data file. This record consists of the time of day, the number of the cell selected on the matrix display, the name of the image that is displayed in the cell at that time, and the text of the speech output or the name of the activated environmental device. The report also includes demographic information on the user, the access mode, the name of the particular matrix display, the dimensions of the matrix, the sensitivity setting on the voice-recognition board, the matrix and scanning box (if one is used), colors, the names of the various images and their cell locations, their associated speech outputs or device activations, and phonetically spelled approximations of the user's voice inputs for the images. When the teacher or user so desires, the software can aggregate and analyze these data on the user's activities and can generate on-screen and/or hardcopy printouts.

APPLICATIONS OF THE SOUND-TO-SPEECH/GRAPHICS SYSTEM

After the development and refinement of the STS/Graphics System, we investigated whether subjects with mental retardation and/or multiple disabilities could learn the cause/effect relationship between vocalization and function. If they learned the cause/effect relationship, could they then learn that different vocalizations are associated with different functions? Would the subjects exhibit increased communication with others as a result of system use? The viability of the graphics component for this population using this type of communication/environmental control system was also an important issue. And finally, what aspects of the system, if any, need further refinement? A manuscript on the behavioral research methods and analysis of the behavioral data is in preparation. A brief summary follows.

Procedures

Two students from a local public school were selected from a pool of twenty-four potential subjects to evaluate the STS/Graphics System. Subject 1 was an eleven-year-old ambulatory female with mental retardation who needed extensive support and who used direct selection to control the system. Subject 2 was a twenty-one-year-old semi-ambulatory female with mental retardation who needed extensive support and who used scanning to control the system.

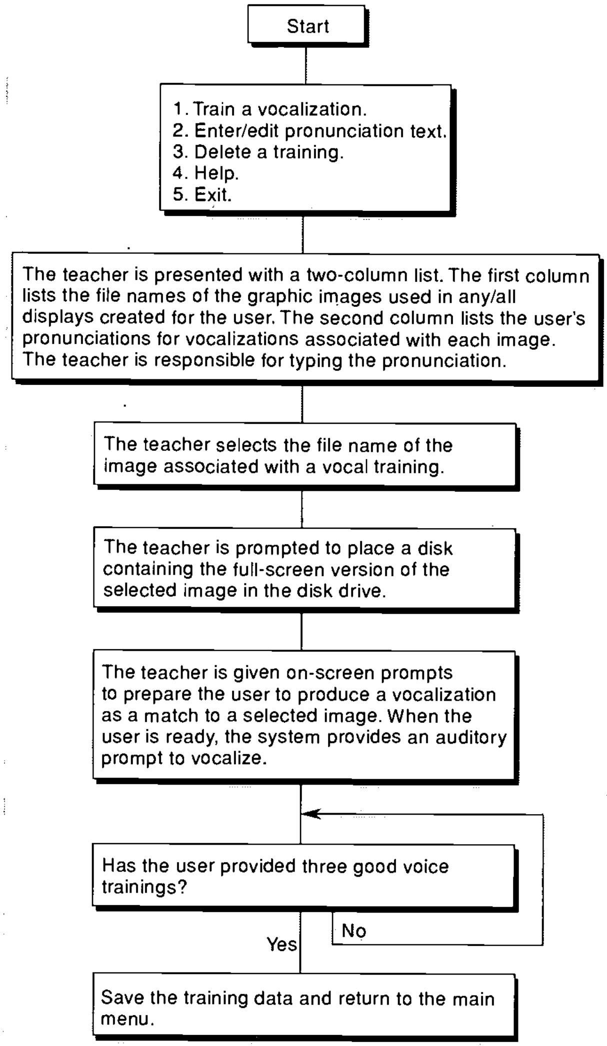


Figure 4. Recognition function of the speech manager component of the Sound-to-Speech/Graphics System software.

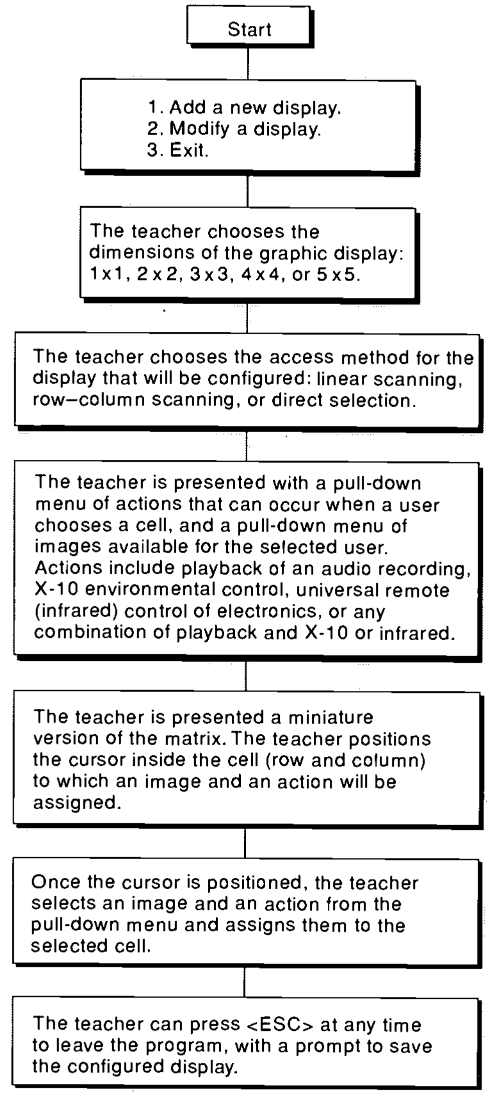


Figure 5. Setup component of the Sound-to-Speech/Graphics System software.

The subjects made few vocalizations, none intelligible; they understood rudimentary scanning, understood cause and effect, had adequate vision and hearing, and were able to sit in an appropriate position to use the STS/Graphics System. Each subject used the STS/Graphics System in her school environment.

Before the introduction of the STS/Graphics System, the subjects were shown photographs of what were assumed to be high-preference items. By using a forced-choice selection procedure with each subject, the photographs were ranked according to the subject's order of preference. The photographs with the highest ranks were scanned into digitized images so that they could be presented on the STS/Graphics System display. The images were linked to spoken communications, environmental control, or a combination of both. Each subject's optimal mode of selection, that is, direct selection or scanning, was also determined. Training strategies were used to teach the subjects how to operate the system using their personal vocalizations. During single-image trials, only one graphics image was displayed on the monitor. Over time, additional graphics images were added to the display, and their corresponding choices were available for activation. Subjects would advance from single-image displays to multiple-image displays when they successfully completed at least nine out of twelve trials on each type of display.

Experimental trials consisted of the researcher orally requesting the subject to activate a specific device or communication phrase, for example, "Tell me about the beads." A trial was scored according to whether the subject emitted the vocalization that corresponded to the requested device or communication, and whether the computer system recognized it as such and activated the proper output. Three levels of prompting could be used with subjects to help them attend to the monitor display and to learn either the direct-selection or scanning strategy. The prompts consisted of verbal instruction, tapping

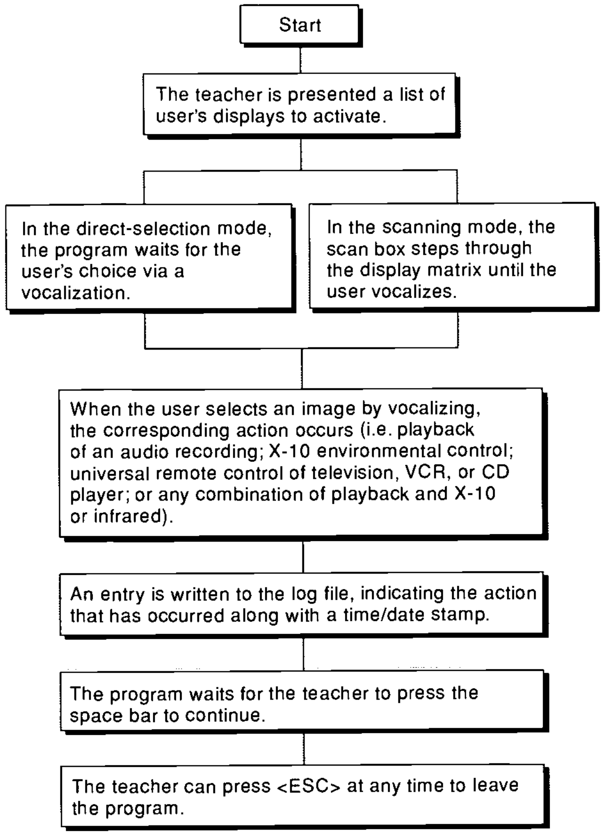


Figure 6. Sound-to-speech component of the Sound-to-Speech/Graphics System software.

on the display, and/or modeling the user's vocalization to aid the subject in learning to control the system.

Subject 1, who used direct selection, progressed through training that began with selecting a single item on the display and ended with selecting from multiple items that were topically similar, for example, beauty aids or food items. Subject 2, who used scanning, did not progress past a single image on a 2 × 2 display at the time that data collection terminated.

Results

The two subjects learned to communicate successfully and to make choices via the STS/Graphics System. They successfully learned the cause/effect relationship between vocalization and action. Subject 1, who used the direct-selection access mode, learned that different vocalizations were associated with different functions. Variability existed between the subjects in their level of proficiency in operating the system. Subject 1 demonstrated an understanding of the use of the system with two choices available, both of which were highly motivating. Her performance deteriorated upon introduction of a third choice. It is unknown if this result was due to boredom, dissatisfaction with the choices available, inability to differentiate the target vocalizations with three choices, or lack of understanding of the use of the system with three choices. Subject 2 demonstrated an understanding of using the scanning access mode to make choices. Although she had multiple items from which to choose, she never advanced beyond a single item on the display at one time because of a lack of time with the system. The graphics component of the system was used appropriately by the subjects in this application.

Aspects of the STS/Graphics System requiring further refinement are (1) facilitating the additional training of the system that sometimes becomes necessary because of the subjects' improved vocalization over time, (2) improving the accuracy of the system's voice recognition in noisy environments, (3) increasing the flexibility of microphone placement, and (4) improving the portability of the system.

CONCLUSIONS

The STS/Graphics System is a viable method of communication and environmental control for children and adults with mental retardation and/or physical disabilities. The system is effective in teaching persons with mental retardation to use their natural speech to communicate and to control their environment, and thereby to become more independent. The graphics interface provides a symbolic visual aid that highlights the options available to users and facilitates making selections. Vocalization is a natural means by which to empower a person, and has proved to be a robust access mode for assistive technology with people having mental retardation. Additional applications of the STS/Graphics System for assessment and educational uses will be explored. Given the enhancement in personal choice-making that the STS/Graphics System provides, as well as its potential to make contributions in other functional areas, future research on the application of voice-recognition technology to the needs of children and adults with mental retardation should be revealing.

REFERENCES

1. "AAMR Board Approves New MR Definition," *AAMR News & Notes* **5**(4), 1,6 (Jul/Aug 1992).
2. "AAMR Issues New Definition of Mental Retardation," *The Arc Today* **41**(4), 3,8 (Jul/Aug 1992).
3. Matas, J., Mathy-Laikko, P., Beukelman, D., and Legresley, K., "Identifying the Nonspeaking Population: A Demographic Study," *Augmentative and Alternative Communication* **1**, 17–28 (1985).
4. American Speech-Language-Hearing Association, *Augmentative Communication for Consumers*, p. 3 (1987).
5. American Speech-Language-Hearing Association, "Augmentative and Alternative Communication," *ASHA* **33**(8), Suppl. 5 (1991).
6. Mirenda, P., and Mathy-Laikko, P., "Augmentative and Alternative Communication Applications for Persons with Severe Congenital Communication Disorders: An Introduction," *Augmentative and Alternative Communication* **5**(1), 3–13 (1989).
7. Locke, P., and Mirenda, P., "A Computer-Supported Communication Approach for a Child with Severe Communication, Visual and Cognitive Impairments: A Case Study," *Augmentative and Alternative Communication* **4**, 15–22 (1988).
8. Romski, M. A., and Sevcik, R. A., "Augmentative and Alternative Communication Systems: Considerations for Individuals with Severe Intellectual Disabilities," *Augmentative and Alternative Communication* **4**(2), 83–93 (1988).
9. Mirenda, P., and Dattilo, J., "Instructional Techniques in Alternative Communication for Students with Severe Intellectual Handicaps," *Augmentative and Alternative Communication* **3**, 143–152 (1987).
10. Blackstone, S., "Equipment: Speech Recognition Systems," *Augmentative Communication News* **5**(1), 1 (1992).

[11] Coleman, C., and Meyers, L., "Computer Recognition of the Speech of Adults with Cerebral Palsy and Dysarthria," *Augmentative and Alternative Communication* 7(1), 34–42 (1991).

[12] Treviranus, J., Shein, F., Haataja, S., Parnes, P., and Milner, M., "Speech Recognition to Enhance Computer Access for Children and Young Adults Who Are Functionally Nonspeaking," in *Proc. RESNA 14th Annual Conf.*, Resna Press, Washington, D.C., pp. 308-310 (1991).

[13] Brown, C. C., and Cavalier, A. R., "Voice Recognition Technology and Persons with Severe Mental Retardation and Severe Physical Impairment: Learning, Response Differentiation, and Effect," *J. Special Education Technol.* XI(4) (Spring 1992).

[14] Brown, C. C., Wyatt, C., Sauer, M., and Cavalier, A., "The Sound-to-Speech Translation Utilizing Graphic Symbols," in *Proc. Sixth Annual Conf. on Technology and Persons with Disabilities*, California State University, Northridge, Calif. (1992).

[15] Brown, C. C., Cavalier, A. R., and Tipton, L., "Increased Independence Through Computerized Voice Recognition for Persons Who Are Severely Physically Involved and Profoundly Mentally Retarded," in *Proc. Ninth Annual Conf. of the Rehabilitation Engineering Society of North America*, Resna Press, Washington, D.C., pp. 101-103 (1986).

[16] Brown, C. C., "Research Focusing on Freedom of Choice, Communication, and Independence Using Eyegaze and Speech Recognition Assistive Technology," in *Proc. First South Central Technology Access Conf.*, University of Arkansas at Little Rock, pp. 27-34 (1989).

ACKNOWLEDGMENTS: Funding for this research was provided by the U.S. Department of Education Grant No. H180P90015, contributors to The Arc Research Fund, and Martin Marietta Corporation. All ideas and information contained in this article do not necessarily reflect the position of the funding sources, and no official endorsement of the funding sources should be inferred. A documentary videotape of this research, *Voices of the Future*, is available from The Arc. A detailed operations and training manual accompanies the STS/Graphics System.

THE AUTHORS

CARRIE BROWN has served as the director of the Bioengineering Program of The Arc (formerly the Association for Retarded Citizens of the United States) for the last three years, where she had previously served as an Assistant Director to the Program since 1985. As Director of the Bioengineering Program, she is directly responsible for its overall administration, direction, and focus. She has been instrumental in the ongoing research tracks of the Program, which are multidimensional but have a special emphasis on the needs of persons with mental retardation and severely handicapping conditions. Her main areas of expertise are the development and field testing of communication and environmental control systems using speech-recognition and eye-gaze technology, software and hardware design, research project management, and the development of training strategies. Dr. Brown holds a doctoral degree in early childhood/special education, a master's degree in special education, and a bachelor's degree in secondary education.

MAGGIE SAUER is the Director of the Assistive Technology Program, Clinical Center for the Study of Development and Learning, at the University of North Carolina. Her responsibilities include evaluation and training of children and adults with disabilities, development of computer-based intervention programs, and consultation to professionals in a variety of settings. Ms. Sauer directs research projects related to assistive technology, as well as pre-service training at the University of North Carolina. She has published and presented numerous papers in the area of assistive technology.

AL CAVALIER is an Associate Professor in the Department of Educational Studies and Director of the Center for Assistive and Instructional Technology at the University of Delaware. He has been active in the fields of special education technology and rehabilitation technology since 1977. As Program Director of the Warrior Center, a large developmental center, he designed and supervised an automated environment for adolescents who are severely mentally retarded and physically impaired. Following this position, he was Director of the Bioengineering Program at the national headquarters of The Arc, where he was involved in the design and evaluation of new assistive technology for persons with cognitive impairments. His current research interests are in human-factors principles involved in the use of assistive technology by persons with disabilities, the effectiveness of computer graphics and synthesized speech as mediators for learning and performance, and the cognitive demands placed on students with disabilities by functional tasks and assistive devices. He is the associate editor of the journal *Computers in Human Services*.

CATHERINE WYATT is the Technology Coordinator for The Arc Bioengineering Program. She provides technical support on technology and software development of all Bioengineering Program research projects. Mrs. Wyatt assists with initial system design, programming, device evaluation, and field testing of program prototypes, and she serves as a technical liaison between The Arc and product designers and engineers (supervising the programming staff). As Technology Coordinator, she researches and evaluates the hardware requirements for all projects and advises on the decisions regarding hardware purchases and interfaces. Before her employment at The Arc, Mrs. Wyatt designed computer systems (hardware, software, and custom software design) for small- to medium-sized businesses.

Presentations Pertaining to the Sound-to-Speech Research Project

1989	**"Assistive Technology Allowing Control and Communication for Users with Severe Handicaps"** - A.I. duPont Institute
	"A Bioengineering Update" - ARC Annual Convention
	"Research Focusing on Freedom of Choice, Communication, and Independence Using Eyegaze and Speech Recognition Assistive Technology" - RESNA Regional Technology Access Meeting
1990	**"Technology for Persons with Mental Retardation: ARC's Bioengineering Program"** - 114th Annual Meeting of the American Association on Mental Retardation
	"Assistive Technology for People with Mental Retardation: The ARC Bioengineering Program" - Closing the Gap Conference
	"Assistive Technology for People with Mental Retardation: The ARC Bioengineering Program" - Southeast Regional American Association for Mental Retardation Conference
	"Bioengineering: Updates and Future Considerations" - 40th Anniversary National Convention of the Association for Retarded Citizens of the United States
	"A Voice Activated Communication System with Graphics" - American Speech-Language-Hearing Association 1990 National Convention
1991	**"Eyegaze and Speech Recognition Technology for People with Severe Handicaps and Mental Retardation"** - Technology and Media International Conference on Special Education Technology.
	"Graphic Interfaces and Speech Output" - Center for Special Education Technology, The Council for Exceptional Children Conference - Technology Seminar on Moderately Handicapped
	"ARC Bioengineering Program Research" - Center for Special Education Technology - Technology Forum for Education Associations
	"Voices of the Future: Sound-to-Speech Translation Utilizing Graphics Interface for Students with Severe Handicaps" - Closing the Gap Conference
	"Assistive Technology for Persons with Mental Retardation" - ARC/Pennsylvania Annual Convention

1992 **"Voices of the Future: Sound-to-Speech Translation Utilizing Graphics Interface for Students with Severe Handicaps"** - Technology and Media Conference

"A Speech Recognition System for Communication, Environmental Control, and Learning Utilizing Graphics Display" - 7th Annual International Conference, Technology and Persons with Disabilities - sponsored by California State University, Northridge

"Sound-to-Speech Translation Utilizing Graphic Symbols" - RESNA International '92 Conference

"Overview of the Bioengineering Program Research" - Thirty-Third Annual Meeting of the Stone Belt Council for Retarded Citizens

"Voices of the Future" (videotape) and **"Speech Recognition and Graphics Research for Persons Having Mental Retardation"** - ISAAC '92 Conference

"The Sound-to-Speech Utilizing Graphics System" - Johns Hopkins National Search for Computing Applications to Assist Persons with Disabilities, Region 6 Regional Fair & National Fair

"Research Findings of a Voice Activated Communication System with Graphics" - American Speech-Language-Hearing Association 1992 National Convention

ASSOCIATION FOR RETARDED CITIZENS
OF THE UNITED STATES

RESEARCH PARTICIPANT OBSERVATION FORM

Project________________________________

Date of Observation -
Name -
Address -
Phone -
Parents Name -

D.O.B. -
Diagnosis-

School-
Address-

Phone-
Teacher-

Persons in attendance

<u>General Information</u>
Medical considerations

Sensory

Motor

Positioning/Seating

Communication

1. general description

2. use of symbol system

3. use of switches

4. use of computer or other technology

Miscellaneous

1. O.T., P.T., SPEECH

2. preferences, nonpreferences

Comments from teacher interview

Recommendations for project participation

Alpha subject ____________________

Beta subject ____________________

Parent followup letter sent____________________

Consent form received for participation____________________

ASSOCIATION FOR RETARDED CITIZENS OF THE U.S.
BIOENGINEERING PROGRAM
EVALUATION-SOUND -TO-SPEECH UTILIZING GRAPHICS

File and Medical Record Review

IDENTIFYING INFORMATION

NAME................:
HOME ADDRESS.......:
HOME PHONE.........:
PARENT(S)...........:

DATE OF BIRTH......:
CHRONOLOGICAL AGE..:

TEACHER(S)..........:
SPEECH PATHOLOGIST.:
SCHOOL NAME........:
SCHOOL ADDRESS.....:
SCHOOL PHONE.......:
SCHOOL DAYS/WEEK...:
SCHOOL HOURS/DAY...:

MEDICAL INFORMATION

MEDICAL DIAGNOSIS..:

SEIZURES?..........:

MEDICATION.........:

VISION.............:

HEARING............:

CURRENT STATUS

PREFERENCES

PEOPLE...........:

FOODS............:

ACTIVITIES.......:

TOYS.............:

AMBULATION

AMBULATORY % ....:
WALKER % ....:
STROLLER % ......:
SCOOTER BOARD %..:
WHEELCHAIR % ....:
WHEELCHAIR TYPE:
POWERED?.......:
CONTROL METHOD:

BEHAVIOR

COMPREHENSION

BY REPORT........:

TEST RESULTS
AVAILABLE........:

SPEECH PRODUCTION

TEST RESULTS....:

BY REPORT.......:

EATING DIFFICULTIES

SPEECH THERAPY (REPORT NUMBER OF SESSIONS/WEEK/TIME, GOALS

HISTORY.........:

PRESENT.........:

COMMUNICATION

GESTURAL........:

BEHAVIORAL INTERPRETATION..:

AIDED COMMUNICATION (COMMUNICATION BOARDS, COMPUTER ETC.)

A.HISTORY

DESCRIBE SYSTEMS USED PREVIOUSLY, NOTE ANY CHANGES OR RATIONALE FOR CHANGE. IN ADDITION NOTE ANY EVALUATIONS SUGGESTING THE USE OF AUGMENTATIVE COMMUNICATION AIDS

B.CURRENT DEVICES:

1. INDICATION (E.G.USE OF FINGER, HAND, EYEGAZE)

2. SYMBOL SET (E.G. PHOTOGRAPHS, LINE DRAWINGS, OBJECTS)

3. SYSTEM USE- DESCRIBE HOW THE PRESENT SYSTEM IS USED, WHEN, WHERE, WITH WHOM, AND HOW EFFECTIVELY IT PROVIDES INDIVIDUAL WITH ABILITY TO COMMUNICATE.

ELECTRONICS

COMPUTER

HOME..........:
TYPE.........:
SOFTWARE.....:
PERIPHERALS..:

SCHOOL........:
TYPE.........:
SOFTWARE.....:
PERIPHERALS..:

SWITCHES

DESCRIBE WHAT TYPE, HOW THEY ARE USED AND HOW THEY ARE ACTIVATED.

SWITCH ACTIVATED TOYS

DESCRIBE TOY AND HOW IT IS ACTIVATED

ACADEMICS

GRADE LEVEL........:
ACADEMIC SYSTEM....:
TYPE OF CLASSROOM..:
CURRICULUM.........:
SPECIAL SERVICES...:
READING LEVEL
RECOGNIZES LETTERS:
RECOGNIZES WORDS..:
READS PHONETICALLY:
SPELLING LEVEL....:

ADDITIONAL COMMENTS

COMMUNICATION :

CONTROL SITE :

SOFTWARE :

EDUCATION :

RECOMMENDATIONS FOR EVALUATION

Regarding communication,

Regarding Materials used for evaluation (e.g. stimulus size, scanning or direct selection, type of materials to use-toys, food, music etc.)

Regarding software,

Regarding Questions for educational staff

Date of Evaluation

IDENTIFYING INFORMATION

Name

Date of Birth

Project

Referral Source

Parent

Home Address

Home Phone

School Name

School Address

School Phone

Teacher(s)

Speech Pathologist

Occupational Therapist

Physical Therapist

Comments:

Location of research project________________________________

Alpha Group______ Beta Group______

Contact Person

Address

Phone

Directions

Research Project
Referral Form

Date:________________

Client/Facility Name____

Address____

Phone____

Age/Diagnosis ____

Contact Person ____

Address: ____

Phone: Daytime__________________Evening__________________

Phone notes: ____

Referred By: ____

Has this person participated in ARC Projects? YES___ NO___

If YES, which project?____

Date of Participation____

PROJECT

	ASSISTIVE DINING DEVICE	STS	EYEGAZE
Referred:	______	______	______
Alpha test	______	______	______
Beta test	______	______	______
Indicated interest in participation	______	______	______

Other

Phone Contact/Interview Date ____________________

Project Consent Form Sent Date ____________________

Project Consent Form Recvd Date ____________________

Follow-up Phone Call Date ____________________

Observation Date ____________________

Evaluation Date ____________________

Selection to Project Date ____________________

DATE	CONTACT: P=PHONE O=OBSERVATION T=TRAINING I=INTERVENTION	SUMMARY

ASSOCIATION FOR RETARDED CITIZENS
OF THE UNITED STATES

RESEARCH PARTICIPANT OBSERVATION FORM

Project________________________________

Date of Observation -
Name -
Address -
Phone -
Parents Name -

D.O.B. -
Diagnosis-

School-
Address-

Phone-
Teacher-

Persons in attendance

General Information

Medical considerations

Sensory

Motor

Positioning/Seating

Communication

1. general description

2. use of symbol system

3. use of switches

4. use of computer or other technology

Miscellaneous

1. O.T., P.T., SPEECH

2. preferences, nonpreferences

Comments from teacher interview

Recommendations for project participation

Alpha subject ______________________________

Beta subject ______________________________

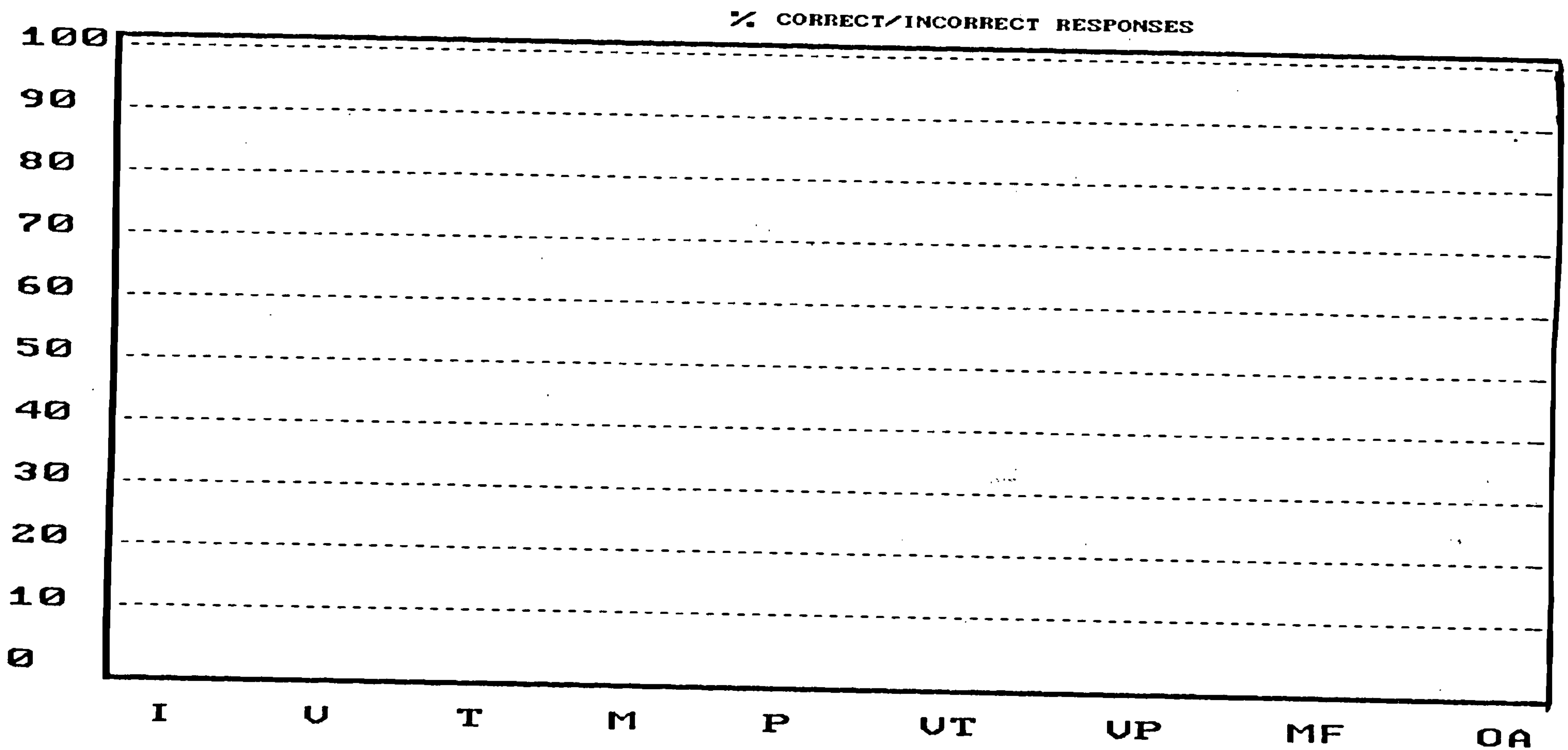
Parent followup letter sent______________________________

Consent form received for participation__________________

312

…pervisor: ____________
…nician: ____________
…dent: ____________
Date: ____________

PROGRESS SHEET



ACTIVITY 1:

ACTIVITY 2:

ACTIVITY 3:

ACTIVITY 4:

ACTIVITY 5:

PROMPT TYPE

I	INDEPENDENT
V	VEREAL
T	TAP
M	MODEL
P	PHYSICAL ASSIST
VT	VERBAL TAP
VP	VERBAL PHYSICAL ASSIST
MF	MISFIRES
OA	OTHER ACTIVATION
SI	SYSTEM INCONSISTENCY
TM	TOUCHING MICROPHONE
SE	SCANNING ERRORS

LINE LEGEND

+	= (double line)
−	= (heavy line)

COMMENTS:

NAME:

	PICTURE	SIZE	X 10	SPEECH	OK
PAGE 1					
PAGE 2					
PAGE 3					
PAGE 4					
PAGE 5					
PAGE 6					
PAGE 7					
PAGE 8					
PAGE 9					
PAGE 10					

PAGE ___ OF ___
N OF PHOTOS: ___

PICTURE FILE NAME: ___
DATE OF EVAL: ___
DIRECT SELECT: ___
SCAN: ___

SCANNED: Y/N
SEX: M/F
AGE: ___

S-T-S MATRIX

SUBJECT: ______________ DATE: ________

A

B

1	2
3	4

C

1	2	3
4	5	6
7	8	9

E

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

F

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

PICTURE NAME	X-10 FUNCTION	Module ID#		SPEECH OUTPUT	ACTUATING SOUND	File Name
(Example: lamp	Light On	A		"I want the light on, please."	"Ahhhhh.")	

BEST COPY AVAILABLE

PAGE ___ OF ___
NUMBER OF PHOTOS: ___

PICTURE FILE NAME: ______
DATE OF EVAL: ______
DIRECT SELECT: ______
SCAN: ______

SCANNED: Y/N
SEX: M/F
AGE: ___

S-T-S MATRIX

SUBJECT: ______________ DATE: __________

A

B

1	2
3	4

C

1	2	3
4	5	6
7	8	9

E

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

F

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

PICTURE NAME	X-10 FUNCTION	Module ID#		SPEECH OUTPUT	ACTUATING SOUND	File Name
(Example: lamp	Light On	A		"I want the light on, please."	"Ahhhhh.")	

Sound-To-Speech Project
Items To Be Photograpned

Date: ________

Item(s): ____

Speech Output: ____

Function: ____

Actuating Sound: ____

Subject
Name: ____

School
Name: ____

School
Adress: ____

School
Phone: ____

Teacher: ____

Alpha Group ________ Beta Group ________________

Location Of Research
Project: ____

Comments:

____

____

____

____

Summary of Daily Progress Reports

Date:______________
Student:___________
Clinician:_________
Supervisor:________

	TR	I	V	T	M	P	VT	VP	MF	OA
Activity 1										
Activity 2										
Activity 3										
Activity 4										
Activity 5										

Key:

I	Independent	VT	Verbal Tap
V	Verbal	VP	Verbal Physical Assist
T	Tap	MF	Misfires
M	Model	OA	Other Activation
P	Physical Assist		-System Inconsistency
			-Touching Microphone
			-Scanning Errors

NOTICE

REPRODUCTION BASIS

- [ ] This document is covered by a signed "Reproduction Release (Blanket)" form (on file within the ERIC system), encompassing all or classes of documents from its source organization and, therefore, does not require a "Specific Document" Release form.

- [x] This document is Federally-funded, or carries its own permission to reproduce, or is otherwise in the public domain and, therefore, may be reproduced by ERIC without a signed Reproduction Release form (either "Specific Document" or "Blanket").